



**Universidade de  
Aveiro**  
**2017**

Departamento de Electrónica,  
Telecomunicações e Informática

**André Filipe Custódio  
Rodrigues**

**Apoio a avaliação analítica em Visualização:  
métodos e uma aplicação**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Informática, realizada sob a orientação científica da Doutora Maria Beatriz Alves de Sousa Santos, Professora Associada com Agregação do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e do Doutor Samuel de Sousa Silva Investigador do Instituto de Engenharia Electrónica e Informática de Aveiro



**o júri**

Presidente: **Professor Doutor Joaquim João Estrela Ribeiro Silvestre Madeira**  
professor auxiliar da Universidade de Aveiro

Arguente Principal: **Professor Doutor Telmo Eduardo Miranda Castelão da Silva**  
professor auxiliar da Universidade de Aveiro

Orientador: **Professora Doutora Maria Beatriz Alves de Sousa Santos**  
professora associada c/ Agregação da Universidade de Aveiro



## **agradecimentos**

A todos as pessoas que directa ou indirectamente me apoiaram durante toda a elaboração deste trabalho. Em especial agradeço aos meus orientador e co-orientador por todo o apoio, acompanhamento e orientação, bem como à minha família e amigos pelo seu incondicional apoio.



<b>palavras-chave</b>	Visualização, avaliação, usabilidade, desenvolvimento centrado no utilizador, aplicação, personas.
<b>resumo</b>	<p>A importância da Visualização de Informação no campo da Informática tem vindo a aumentar ao longo dos últimos anos, tornando-a num tema cada vez mais pertinente. Apesar disso, ainda existem muito poucas aplicações e ferramentas que permitam o uso dos métodos que podem ser utilizados para avaliar soluções ao nível da visualização, o que as torna difíceis de incorporar no processo de desenvolvimento. Através do desenvolvimento de ferramentas para suportar os métodos de avaliação é possível facilitar essa integração e consequentemente promover uma melhoria significativa na usabilidade de aplicações de visualização.</p> <p>Esta dissertação apresenta uma breve introdução à Avaliação em Visualização e métodos de avaliação que podem ser usados nesta área e segue uma abordagem centrada no utilizador no desenvolvimento de uma aplicação para apoiar a avaliação analítica de soluções de visualização. O trabalho resultou num processo de desenvolvimento e avaliação iterativos de protótipos e, na sua forma actual, a aplicação já disponibiliza um subconjunto das funcionalidades pretendidas e foi validada através de testes de usabilidade com um conjunto de utilizadores.</p>





<b>keywords</b>	Visualization, evaluation, usability, user-centered design, application, personas.
<b>abstract</b>	<p>The importance of Information Visualization in the field of Computing has been steadily increasing over recent years, making an ever more pertinent topic. Despite that, there still exist very few tools or applications allowing an efficient use of the methods that can be considered to evaluate visualization solutions, which in turn makes evaluation hard to incorporate in the software development process. Through the development of tools to support evaluation methods used in Visualization it is possible to ease this integration and consequently foster a significant improvement in the usability of future visualization applications.</p> <p>This dissertation presents a brief introduction to Evaluation in Visualization, its methods as well as showcasing the advantages of using a user-centered approach to the development of an application to support the analytical evaluation of Visualization solutions. The work carried out resulted in the iterative development and evaluation of application prototypes and, at its current stage, the application already provides a subset of the intended features and has been validated through usability testing by a group of users.</p>



## Índice

Índice.....	i
1 – Introdução.....	1
1.1 – Motivação.....	1
1.2 – Objectivos.....	2
1.3 – Visão Geral .....	3
2 – Trabalho Relacionado .....	5
2.1 – Avaliação em Visualização.....	5
2.1.1 – Avaliação Heurística .....	6
2.1.2 – Experiências Controladas .....	8
2.1.3 – Questionários.....	9
2.2 – Objectivos e Desafios da Avaliação em Visualização .....	10
2.3 – Soluções existentes para apoiar a avaliação de Interfaces de utilizador (IUs).....	11
2.3.1 – Crescente Utilização da Visualização .....	11
2.3.2 – Análise Ambiental .....	16
2.4 – Desenvolvimento Centrado no Utilizador e Personas .....	19
2.4.1 – Desenvolvimento Centrado no Utilizador .....	20
2.4.2 – Personas .....	21
3 – Análise de Requisitos e Modelo Conceptual.....	23
3.1 – Definição do público-alvo.....	23
3.2 – Personas Criadas.....	25
3.3 – Modelo conceptual da aplicação .....	31
3.4 – Evolução do protótipo.....	31
3.4.1 – Primeira Versão Conceptual .....	32
3.4.2 – Segunda Versão Conceptual .....	32
3.4.3 – Versão Actual do Protótipo Funcional.....	33
3.5 – Evolução e Prototipagem das Interfaces de Utilizador .....	33
3.5.1 – Protótipo de Baixa Fidelidade.....	34
3.5.2 – Teste de Usabilidade do protótipo.....	35
4 – Implementação, Avaliação e Resultados .....	43
4.1 – Linguagem utilizada e considerações sobre a implementação .....	43

4.1.1	– Balsamiq Mockups 3.....	43
4.1.2	– Linguagem C#.....	44
4.1.3	– Microsoft Visual Studio 2017 .....	44
4.2	– Implementação.....	45
4.3	– Avaliação da aplicação .....	51
4.4	– Discussão dos Resultados.....	53
5	– Conclusões e Trabalho Futuro.....	55
	Referências.....	57
	Anexos.....	60

## Lista de Figuras:

Figura 1– Aspecto da Interface de utilizador da aplicação Visage (Roth, 1996) .....	12
Figura 2– Interação entre múltiplas janelas na aplicação Visage. Dragging das barras de um gráfico para o mapa (Roth, 1996) .....	12
Figura 3– <i>Drag and Drop</i> no Quant-Ux .....	14
Figura 4–Ferramenta de Análise de Interface Quant-Ux “ClickTracker” .....	15
Figura 5– Hierarquização de Aplicações de avaliação de Usabilidade.....	16
Figura 6–Silverback .....	17
Figura 7–Feng- GUI .....	18
Figura 8–Try My UI.....	19
Figura 9 - Levantamento de Público-alvo e Requisitos .....	24
Figura 10 - Menu Inicial da Aplicação .....	36
Figura 11 - "Project Selection".....	37
Figura 12 - Interface Principal da Aplicação .....	38
Figura 13 - Interface Alternativa do Menu Principal.....	39
Figura 14 - Menu de Partilha .....	40
Figura 15 - Ecrã inicial para escolha de uma das modalidades de avaliação previstas: Experiências Controladas ou Avaliação Heurística .....	45
Figura 16- Ecrã do protótipo do gestor de projectos do utilizador .....	46
Figura 17 - Menu Principal na sua forma mais básica sem heurísticas ou imagem.....	47
Figura 18 – Menu Principal após adicionar heurísticas e imagem.....	48
Figura 19 - Menus relacionados com a escolha do grupo de heurísticas, no topo, e a lista das heurísticas que o compõem. Neste caso, em particular, foram seleccionadas as heurísticas de Nielsen.....	49
Figura 20 – Conjunto de opções disponíveis para um problema assinalado .....	50



# 1 – Introdução

Actualmente a usabilidade é um tópico cada vez mais relevante. Com o elevado número de aplicações disponíveis, muitas delas inclusive com o mesmo fim, o primeiro impacto no utilizador pode determinar o seu sucesso ou fracasso. Estas aplicações, especialmente as que apresentam um grande volume de dados aos utilizadores de forma visual, devem ser desenvolvidas desde início com objectivos precisos no que se refere à sua usabilidade por parte dos futuros utilizadores.

## 1.1 – Motivação

A avaliação de sistemas de Visualização de Dados ou Informação está sujeita a um conjunto de desafios específicos que a diferenciam, em alguns aspectos, das avaliações efectuadas a outro tipo de aplicações interactivas. Esta diferenciação obriga a que os métodos considerados tenham em conta essas especificidades e a literatura tem avançado algumas propostas de métodos analíticos de avaliação para esse contexto (Carpendale, 2008)(Forsell et al., 2010)(Santos et al.,2016).

A investigação da adequabilidade de diferentes métodos analíticos, nomeadamente aqueles que se baseiam na aplicação de listas de heurísticas, tem procurado propor e validar listas de heurísticas que possam auxiliar os utilizadores numa avaliação inicial das suas visualizações (Zuk et al., 2006) (Forsell et al., 2010). No entanto, na falta de ferramentas que permitam tornar a aplicação de avaliação heurística mais estruturada

e expedita, a recolha de dados experimentais para suportar essa investigação encontra-se dificultada. Para além disso, a falta deste tipo de ferramentas é também uma barreira à consideração destes métodos de forma mais sistemática por parte da comunidade que lida com o desenho e desenvolvimento de aplicações.

## 1.2 – Objectivos

Este trabalho tem como principal área de aplicação a Visualização de Dados e Informação e pretende auxiliar nos processos de avaliação durante desenvolvimento de aplicações de Visualização tornando mais expedita a sua utilização e assim contribuindo para um melhor resultado quanto a usabilidade e UX (*User Experience*).

O objectivo principal desta dissertação consistiu em estudar métodos usados tradicionalmente na avaliação de sistemas interactivos que fossem facilmente adaptáveis para avaliar aplicações de Visualização e desenvolver uma aplicação para apoiar a aplicação destes métodos nesse contexto. Como mencionado, pretende-se que esta aplicação seja específica para a área da Visualização, não pretendendo competir com outras aplicações mais genéricas para apoio a avaliação de usabilidade existentes no mercado.

O estudo dos métodos de avaliação de usabilidade mais frequentemente usados permitiu a identificação da avaliação heurística, bem como dos questionários e experiências controladas como potencialmente úteis na avaliação em Visualização e adequados a inclusão na aplicação. Ao nível da avaliação heurística tornou-se evidente a necessidade de dotar a aplicação de listas de heurísticas diversas que facilmente sejam associáveis a imagens ou vídeo, bem como diferentes formas de realçar e classificar potenciais problemas e proporcionar *feedback* relativo aos problemas, incluindo funcionalidade para o auxílio na escrita de relatórios correspondentes a avaliações heurísticas de forma mais automatizada. A aplicação foi desenvolvida usando uma abordagem iterativa centrada no utilizador, incluindo vários momentos de avaliação de protótipos com diferentes graus de fidelidade.

Esta dissertação pretende assim dar um contributo para o desenvolvimento mais sustentado de sistemas de Visualização de Dados e Informação, tentando tornar o processo de avaliação um pouco mais fácil para quem nela trabalha e esperando, em ultima instância, auxiliar na criação de melhores visualizações.



## 1.3 – Visão Geral

No Capítulo 2 desta dissertação é apresentada uma introdução à avaliação em Visualização descrevendo os seus principais objectivos bem como a sua importância e os principais métodos existentes para atingir esses objectivos.

Descrevem-se também aplicações existentes para apoiar avaliação de usabilidade de interfaces de utilizador que, não sendo específicas para aplicações de Visualização, proporcionam ideias que podem ser úteis nesse contexto.

No Capítulo 3 definem-se objectivos, Personas, cenários, requisitos e testes dos diferentes protótipos em papel para a validação do modelo conceptual duma aplicação para apoio à avaliação em Visualização.

O Capítulo 4 apresenta uma breve discussão dos principais aspectos da implementação da aplicação bem como uma avaliação feita à aplicação, incluindo ainda uma análise crítica aos resultados obtidos.

A dissertação é concluída no Capítulo 5 onde são apresentadas conclusões bem como sugestões para trabalho futuro que pode ser desenvolvido com base no trabalho realizado no âmbito desta dissertação.



## 2 – Trabalho Relacionado

Sendo esta dissertação centrada na Avaliação em Visualização, é essencial fornecer uma introdução ao tema, passando pelos seus objectivos, principais métodos e ferramentas existentes, bem como os principais desafios enfrentados.

### 2.1 – Avaliação em Visualização

A avaliação de usabilidade é uma área de extrema importância para o desenvolvimento de sistemas interactivos que são utilizados por uma grande variedade de utilizadores com diferentes perfis e em diferentes contextos. Infelizmente é muitas vezes descurada em fases iniciais do desenvolvimento o que pode levantar constrangimentos numa fase mais avançada quando as alterações já são mais difíceis de efectuar e onde o seu custo é bastante mais elevado (Mayhew, 1999).

Também no contexto do desenvolvimento de aplicações de Visualização a avaliação é fundamental, tendo como objectivo garantir também a “qualidade” das visualizações usadas nas aplicações e não apenas a usabilidade da componente de interacção (Freitas et al., 2002)(Freitas et al., 2009).

Tal como outras áreas, também a avaliação em Visualização possui métodos específicos para cumprir os seus objectivos. Apresentam-se em seguida alguns dos mais utilizados e que são relevantes no contexto desta dissertação. Salienta-se ainda a existência de outros métodos que não irão ser mencionados em detalhe (Carpendale , 2008) por não serem suportados pela aplicação, pelo menos nesta fase inicial.

Os métodos mais utilizados em Interação Humano-Computador podem também ser usados em Visualização e podem dividir-se em dois grupos: métodos empíricos e métodos analíticos.

Os métodos empíricos envolvem directa ou indirectamente utilizadores para obter *feedback* e directivas sobre variados aspectos de aplicações (Dix et al., 2004). Alguns exemplos de métodos empíricos são questionários a utilizadores, observação directa dos seus comportamentos ou testes de usabilidade que combinam os dois métodos. Outro exemplo de método empírico são as experiências controladas, que em geral envolvem várias tarefas e medidas preparadas por um investigador com o intuito de avaliar algo específico e que são utilizadas no desenvolvimento da área científica da Visualização para confirmar hipóteses.

Os métodos analíticos são métodos que não envolvem interação directa com utilizadores. Nestes incluem-se métodos como *cognitive walkthrough* e avaliação heurística métodos baseados em revisão e avaliação com base em heurísticas (Dix et al., 2004). Neste caso particular as heurísticas tomam a forma de regras de boas práticas, normalmente agrupadas em determinado tema e que devem ser cumpridas quando se desenvolve uma aplicação no âmbito desse tema. Existem vários tipos de heurísticas, desde heurísticas genéricas como as de Nielsen (Nielsen, 1993) até heurísticas específicas para avaliação em Visualização (Zuk et al., 2006) (Forsell et al., 2010). Estes métodos são relativamente simples e podem ser utilizados em fases iniciais do projecto para identificar problemas que possam ser corrigidos antes de testes de usabilidade ou experiências controladas que, envolvendo utilizadores, têm uma logística complexa.

Durante a realização desta dissertação decidiu-se que iriam ser focadas avaliação heurística bem como as experiências controladas. Mais tarde durante o processo de desenvolvimento da aplicação, e devido a restrições de tempo e à necessidade de desenvolver uma das componentes da forma mais completa possível, a avaliação heurística passou a ser o foco principal sendo o único método suportado pela versão actual do protótipo.

### 2.1.1 – Avaliação Heurística

A avaliação heurística é um método de inspecção de usabilidade, primariamente utilizado na área de computação para identificar problemas relacionados com a interface de utilizador. Envolve a avaliação da interface por parte de avaliadores que irão julgar a sua concordância com vários princípios de usabilidade conhecidos como heurísticas (Nielsen, 1993) (Dix et al., 2004).

Uma avaliação heurística<sup>1</sup> é realizada fazendo com que cada avaliador individual inspecione a interface individualmente à procura de problemas de usabilidade ou aspectos menos bem conseguidos, tendo por norma critérios de avaliação pré-definidos (Nielsen, 1994). Esses problemas são classificados com base na sua severidade, sendo a escala de Nielsen<sup>2</sup> uma das mais utilizadas. Esta escala de severidade baseia-se em três aspectos principais: a frequência do problema, o seu impacto e a sua persistência, existindo uma escala de 0 a 4 que os classifica posteriormente e que vai desde problema cosmético a catástrofe de usabilidade. Após todas as avaliações individuais estarem completas é permitido aos avaliadores comunicarem entre si e agregarem as suas descobertas. Os resultados são posteriormente agregados num relatório formal ou relatados a um observador enquanto o avaliador navega na interface.

A simplicidade de uma avaliação heurística torna-a extremamente benéfica nas fases iniciais do desenvolvimento de aplicações, podendo ser utilizados vários conjuntos de heurísticas com propósitos diferentes que podem ser utilizados para fazer avaliações bastante completas e fiáveis já que esses conjuntos não são mutuamente exclusivos. Também o facto de não necessitar de utilizadores torna este tipo de avaliação relativamente atractiva tanto a nível logístico como económico.

A principal desvantagem deste método de avaliação é a subjectividade inerente ao mesmo. Apesar de poder ser feito com apenas um avaliador, a presença de vários é a única forma de garantir resultados relativamente sólidos sem influência da subjectividade. Também a experiência dos avaliadores é um factor determinante na avaliação e nos resultados produzidos pela mesma.

Existem propostos na literatura diferentes conjuntos de heurísticas. As heurísticas de Nielsen são as mais utilizadas pois são mais gerais e assim abrangendo vários aspectos, sendo que são ainda relativamente fáceis de utilizar (Nielsen, 1993). Outras heurísticas têm sido propostas para expandir o âmbito de aplicação das heurísticas de Nielsen ou para cobrir aspectos mais específicos de alguns domínios de aplicação (Hermawati et al., 2016) como por exemplo aplicações móveis (Gómez et al., 2014), e até mesmo crianças com Perturbações do Espectro Autista (Khowaja et al., 2015).

No que respeita à avaliação heurística de sistemas de visualização, Zuk et al. (2006) propõe um conjunto de heurísticas focadas na percepção de cor e estética, na componente cognitiva da integração e compreensão e finalmente na componente usabilidade através da consistência e *feedback* para o utilizador. Forsell et al. (2010) propõe um também conjunto de heurísticas que pretende juntar as heurísticas mais importantes de Nielsen e de outros autores de modo a

---

<sup>1</sup> <https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/> (Acedido em 1-08-2017)

<sup>2</sup> <https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/> (Acedido em 1-08-2017)

criar um novo conjunto de heurísticas, genérico de fácil utilização e que seja mais abrangente que o originalmente proposto por Nielsen.

A avaliação heurística é uma componente de extrema importância neste trabalho sendo um dos dois métodos de avaliação que se pretendem suportar com esta aplicação.

## 2.1.2 – Experiências Controladas

As experiências controladas são uma aproximação que tem sido adoptada por métodos de investigação da psicologia e outras áreas científicas e que tem grande proeminência nos métodos de investigação em Interação Humano-Computador. São muito utilizadas na avaliação de interfaces de utilizador, estilos de interacção e na percepção de conhecimento no contexto de interacção com sistemas. Em geral, pretendem dar resposta a perguntas do tipo “Mudanças na variável X terão efeitos significativos no valor de Y?” (Blandford et al., 2008).

Para a utilização deste método existem várias variáveis que devem ser consideradas: *design* da experiência, participantes, material necessário e procedimento.

A experiência pode ser realizada usando um *design* experimental “Dentro de Grupos” onde cada participante realiza todas as tarefas dentro das condições previstas enquanto uma experiência “Entre Grupos” envolve que o utilizador apenas realize ao abrigo de uma das condições estabelecidas (Blandford et al., 2008). A utilização da metodologia “Dentro de Grupos” permite reduzir o número de inconsistências baseadas na individualidade dos participantes mas introduz o problema de condições anteriores podem condicionar os resultados de tarefas seguintes. A metodologia “Entre Grupos” tem como principal vantagem a não influência dos resultados por factores externos, mas tem a desvantagem de necessitar de um número elevado de participantes e de ser uma metodologia complexa e sujeita à influência do perfil dos participantes (Dix et al., 2004).

Ao nível dos participantes é importante a consideração de qual a população de utilizadores adequada à experiência. Por exemplo, se a experiência for designada para otimizar uma tarefa de uma área muito específica é importante que os participantes da experiência sejam pessoas com experiência nessa área. Também deve ser considerada qual a faixa etária pretendida e utilizar participantes que representem essa faixa etária. Apesar de simples estas directivas são muitas vezes ignoradas face à muito maior facilidade de incorporar por exemplo estudantes e, neste caso, essa fraca representação deve ser tida em conta. Finalmente é necessário ter algum cuidado com considerações éticas, nomeadamente ao nível de participantes vulneráveis (idosos, crianças, etc.), participação através de consentimento informado por parte dos utilizadores e precauções ao nível da salvaguarda da confidencialidade e privacidade dos mesmos (Blandford et al., 2008).

O *design* de uma experiência controlada deve ter em atenção que o objectivo da mesma é testar uma hipótese, normalmente que efeito a mudança de um factor (variável independente) terá nos restantes. Assim, um dos desafios de desenhar uma experiência controlada está em minimizar a hipótese de existirem variáveis que variem entre experiências de utilizadores diferentes, já que as experiências devem ser o mais uniforme possível. Estas variáveis existem inclusive nos utilizadores sendo que sexos diferentes ou pessoas com diferente nível de educação podem criar variações indesejadas.

O material utilizado na maioria das experiências relacionadas com Interação Humano-Computador envolve normalmente um computador com determinado *software*. Assim é importante também aqui uniformizar para que, caso a experiência seja realizada em vários computadores estes possuam características semelhantes, nomeadamente ao nível do sistema operativo, no entanto podem existir outras variáveis dependendo da experiência como, por exemplo, a qualidade de som.

Por fim, deve ser criado um procedimento formal que descreve o que os participantes devem fazer durante a experiência. Em primeiro lugar isto permite ter a certeza que todos os utilizadores que irão realizar a tarefa irão seguir os mesmos passos, criando assim maior uniformização e em segundo lugar permite a outros investigadores replicar a experiência e assim realizar uma maior investigação científica por várias pessoas se tal se verificar necessário para confirmar a hipótese em estudo.

As experiências controladas possuem algumas vantagens, nomeadamente o facto de os seus resultados poderem estabelecer relações de causa, ou seja, que é possível determinar as relações e efeitos entre as diversas variáveis, permitindo também o estudo profundo de variáveis individuais enquanto todas as outras permanecem constantes. A principal desvantagem das experiências controladas é o facto de serem normalmente realizadas em laboratório em condições controladas, sendo assim artificiais e eliminando muitos dos efeitos reais que apenas se manifestam em ambientes reais (Dix et al., 2004).

### 2.1.3 – Questionários

Um questionário é um instrumento de pesquisa constituído por várias questões com o propósito de obter informações, por norma para tratamento estatístico ou para recepção de *feedback* (Dix et al., 2004).

Os questionários possuem vantagens em relação a outros tipos de inquérito (como as entrevistas), nomeadamente o facto de serem mais baratos e por norma serem de fácil preenchimento do que os restantes métodos abordados, auxiliado ainda pelo facto de existirem muitas vezes respostas normalizadas que podem ser utilizadas. Outra grande

vantagem da utilização deste método é a sua capacidade de atingir muitos utilizadores num curto intervalo de tempo, mantendo o anonimato.

Apesar destas vantagens, os questionários têm também algumas desvantagens, como o facto das respostas normalizadas serem muitas vezes frustrantes para os utilizadores respondentes. Em geral são compostos por um pequeno número de questões o que, aliado à subjectividade do método, torna limitada a quantidade e por vezes a qualidade da informação obtida. É também impossível incluir certos grupos de pessoas, como por exemplo, os analfabetos.

Ao longo deste trabalho utilizaram-se questionários, durante as várias fases de teste da aplicação, nomeadamente em testes com utilizadores quando foi necessário obter *feedback*. Também a componente de experiências controladas que poderá ser desenvolvida como trabalho futuro deve contemplar a inclusão de ferramentas de construção de questionários como forma de obtenção de *feedback* dos participantes.

## 2.2 – Objectivos e Desafios da Avaliação em Visualização

A grande variedade de métodos de avaliação ao dispor da Visualização torna-a uma área extremamente versátil capaz de se adaptar às mais variadas situações. Ainda que possa ser aplicada numa grande variedade de situações, a avaliação de Visualizações, de acordo com Plaisant (2004) é aplicada sobretudo nas seguintes:

- Estudos com utilizadores para obter directivas de *design*, onde os utilizadores auxiliam na criação de novos métodos e técnicas destinadas a determinada visualização com vista à sua melhoria, o que tem como consequência uma melhoria na sua usabilidade.
- Avaliação de aplicações em laboratório onde existe uma visualização já disponível e onde, por norma, são utilizadas heurísticas para avaliar e dar *feedback* com vista à sua melhoria.
- Testes de usabilidade com vista a obtenção de *feedback* qualitativo que ajuda na melhoria de experiências interactivas.
- Estudo de ferramentas em ambientes realistas é outra aplicação essencial da Avaliação em Visualização já que permite avaliar se uma determinada funcionalidade, interface ou visualização se adequariam ao ambiente onde se pretende que sejam inseridas. Alguns aspectos a considerar nestas análises são a aceitação por parte dos utilizadores e a adequação de funcionalidades às suas necessidades.

É assim possível ter uma ideia da importância e relevância da Avaliação em Visualização, bem como do seu contributo para a criação e validação das mais variadas aplicações. Mas, apesar dos contributos proporcionados, existem também desafios a enfrentar, sendo alguns deles bastantes específicos e de difícil resolução (Plaisant, 2004).



Em primeiro lugar existe uma necessidade e procura constante por melhores métodos de avaliação. Apesar de já bastante desenvolvidos, os testes, heurísticas, experiências controladas, etc. devem sempre acompanhar a evolução do meio onde estão a ser utilizados, algo que nem sempre é imediatamente conseguido, o que pode dar origem a problemas no momento da sua utilização.

É também importante a conciliação das ferramentas, utilizador e a situação que os envolve. Será esta ferramenta melhor que as outras? Porquê? Em que situações? Para que utilizadores? Será que vão utilizá-la em vez das já existentes? Dá resposta às suas necessidades? Existe portanto uma grande variedade de preocupações e questões que têm de ser respondidas para tentar conciliar esses três elementos, o que não tende a ser fácil ou óbvio.

Finalmente chega-se ao recorrente problema da usabilidade: comprovar melhorias. Apesar de uma aplicação ter sido avaliada e modificada com recurso a técnicas de visualização é impossível afirmar, com certeza, que existem melhorias significativas ao nível da usabilidade que tenham necessariamente sido fruto das alterações realizadas. Podem existir fortes indícios ou suspeitas dessas relação mas é difícil comprová-la.

Estes são assim alguns dos desafios que se pretende investigar e atenuar com a realização desta dissertação.

## **2.3 – Soluções existentes para apoiar a avaliação de Interfaces de Utilizador (IUs)**

No início deste trabalho já se conhecia a existência de várias aplicações para o apoio a avaliação de interfaces de utilizador. Estas, no entanto, eram aplicações genéricas, não tendo sido possível encontrar nenhuma aplicação especificamente vocacionada para avaliação em Visualização. Apesar disso, encontraram-se aplicações que permitiram obter ideias aplicáveis ao desenvolvimento da aplicação aqui apresentada.

### **2.3.1 – Crescente Utilização da Visualização**

Em primeira instância é importante perceber o paradigma para o qual devemos desenvolver. Para obter essa informação foi feita pesquisa bibliográfica por artigos científicos pois são uma fonte mais credível, com resultados comprovados e justificados, e a partir dos quais é possível obter uma noção de evolução de ideias. Um artigo em particular, de Catherine Plaisant (2004), descreve a existência de uma mudança de paradigma em visualização. De acordo com a autora, as visualizações deixariam de ser feitas em laboratório e de forma restrita e desenvolvidas

especificamente para uso de algumas pessoas, para passarem a ser integradas de forma mais natural em aplicações relevantes. Também o seu desenvolvimento passaria a envolver o utilizador como resultado dessa mudança o que tornaria mais fácil a utilização dessas aplicações por utilizadores não especializados.

Seguidamente, descrevem-se brevemente algumas aplicações com o objectivo não só de demonstrar a concretização da previsão da autora bem como o estado actual do mercado de Visualização. A escolha destas aplicações prendeu-se com o facto de ambas estarem a ser apresentadas durante a fase de desenvolvimento, o que mais facilmente mostra a forma como os seus criadores vêem o mercado em que as aplicações se devem inserir e, estando em situação semelhante é mais fácil compará-las.

Refere-se em primeiro lugar a aplicação Visage (Roth, 1996), apresentada em 1996, e que tinha como principal objectivo criar uma interface que permitisse facilitar a gestão de problemas complexos. Esta aplicação pretendia coordenar a visualização de múltiplas variáveis através de uma interface de utilizador relativamente complexa, onde essas variáveis eram representadas em várias janelas já criadas para o efeito. Era, no entanto, possível alguma flexibilidade na forma como eram visualizados os dados, existindo funcionalidade de *Drag and Drop* entre as janelas. As figuras 1 e 2 mostram exemplos da interface de utilizador do Visage.

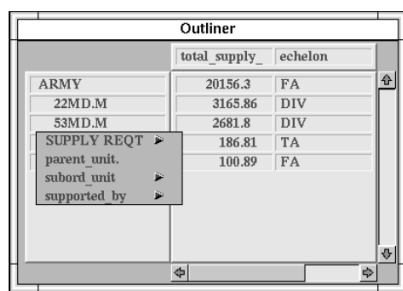


Figura 1– Aspecto da Interface de utilizador da aplicação Visage (Roth, 1996)

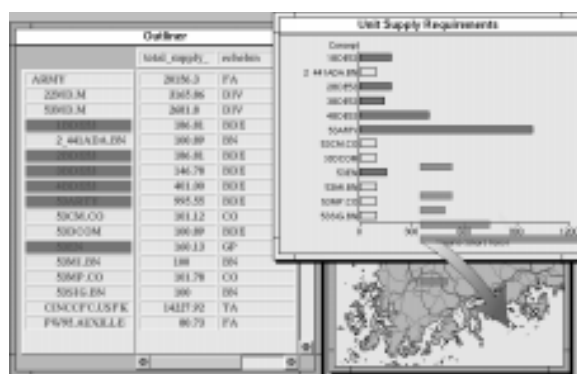


Figura 2– Interação entre múltiplas janelas na aplicação Visage. *Dragging* das barras de um gráfico para o mapa (Roth, 1996)

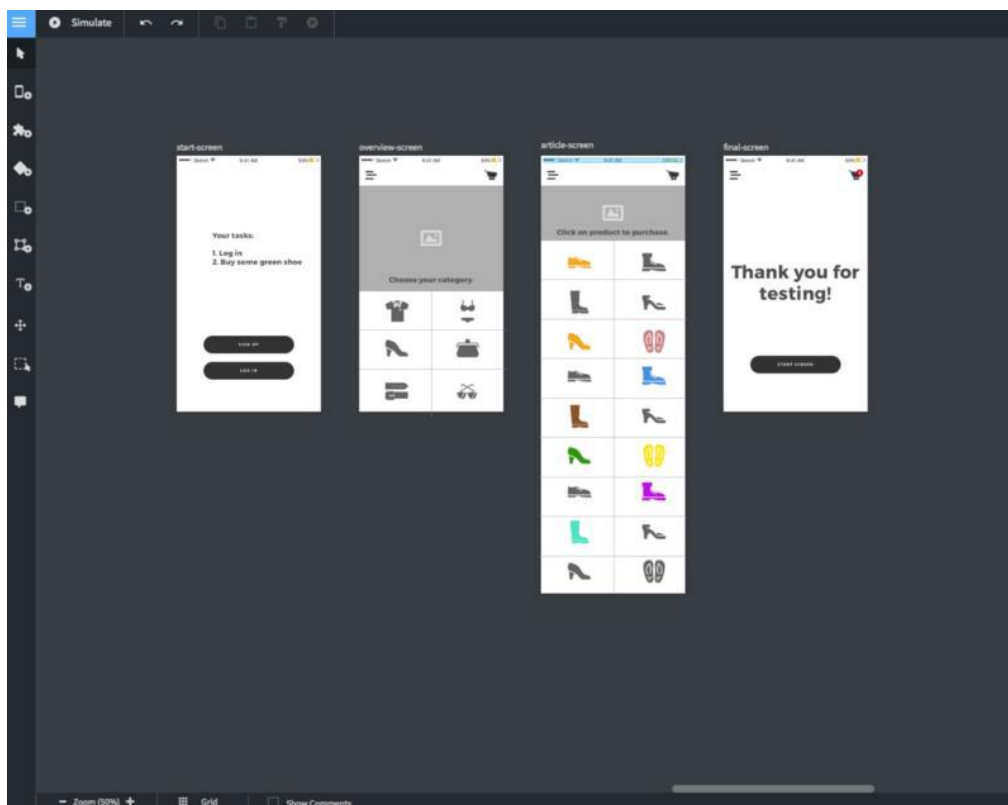
No artigo, são demonstradas aplicações possíveis do Visage em Logística, tentando resolver o problema da gestão de complexidade inerente a essa área. Pensada para ser utilizada por um grupo restrito de pessoas, esta aplicação tinha uma curva de aprendizagem relativamente grande e as interfaces de utilizador não eram claras ou intuitivas para utilizadores fora desse grupo. A tentativa de adaptar vários tipos de visualizações à área da logística (desde transportes a inventários) foi a principal razão da sua complexidade, tendo sido um projecto desenvolvido num ambiente aparentemente fechado, encaixando perfeitamente na descrição de aplicação “feita em laboratório” conforme mencionado por Catherine Plaisant (2004).

Em claro contraste com o Visage encontra-se a aplicação Quant-Ux<sup>3</sup> que se encontrava ainda em desenvolvimento em finais de 2016, e que é exemplo de uma tentativa de integração da visualização de informação nos processos de desenvolvimento modernos e tenta apoiar um utilizador menos especializado.

Apesar de não ser específica para avaliação em Visualização o Quant-Ux é uma ferramenta de criação e análise de interfaces de utilizador, possuindo um sistema de *Drag and Drop*, como pode ser visto na figura 3. Qualquer utilizador pode criar as suas interfaces, não sendo necessário conhecimentos muito específicos de programação, pelo que é relativamente simples criar interfaces interactivas.

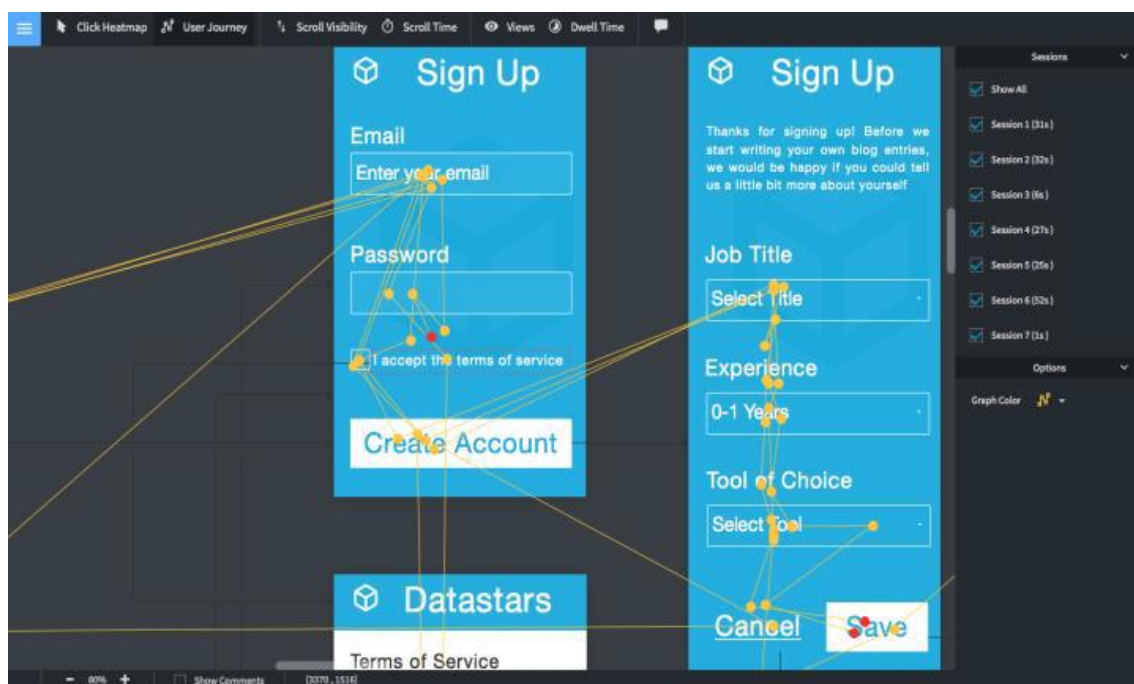
---

<sup>3</sup> <https://www.quant-ux.com/> (Acedido em 1-08-2017)



**Figura 3—Drag and Drop no Quant-Ux**

Outra funcionalidade oferecida pela aplicação Quant-Ux é a existência de várias ferramentas de análise da interface de utilizador incluindo a criação de *heat maps* e possibilidade de ver o percurso do cursor do utilizador na interface, verificando as suas acções e a forma como vai navegando, como pode ser observado na Figura 4. São ferramentas de utilização simples e que facilmente permitem uma análise dos dados recolhidos, possibilitando algum teste e *feedback* integrado num ambiente de desenvolvimento ao alcance do utilizador comum.



**Figura 4—Ferramenta de Análise de Interface Quant-Ux “ClickTracker”**

Uma das características mais interessantes do Quant-Ux é a sua filosofia. O Quant-Ux pretende ser integrado no processo de desenvolvimento de aplicações, tentando que as empresas passem a criar as suas interfaces de utilizador através da aplicação, sendo essas interfaces depois exportadas e utilizadas noutras ferramentas de produção utilizadas pela empresa. No entanto, existem alguns problemas que devem ser referidos. Por um lado, a ausência de tutoriais torna frustrante a sua utilização, especialmente em funcionalidades avançadas. Por outro lado, e mais relevante, é a resistência à mudança por parte dos utilizadores, sendo que as empresas possivelmente preferem continuar a utilizar as ferramentas com que já estão familiarizadas.

Apesar de não serem necessariamente aplicações específicas de avaliação em Visualização, quando conjugadas com a importância dada ao utilizador e do seu envolvimento na evolução das aplicações antes, durante e após o seu desenvolvimento, exemplificam a mudança de paradigma mencionada por Catherine Plaisant (2004), algo que será importante durante o desenvolvimento da aplicação apresentada nesta dissertação.

## 2.3.2 – Análise Ambiental

Após estabelecido o paradigma de desenvolvimento, é necessário perceber o mercado onde a aplicação está inserida de modo a antecipar a reação do utilizador e como se vai posicionar em relação às restantes. Para este fim foram analisados dois *websites*<sup>4</sup>, escritos em formato de *top-lists*, um contendo 24 aplicações maioritariamente relacionadas com a avaliação de interfaces de utilizador, sendo o outro um *follow-up* de apenas 14 em algumas estavam presentes no anterior e outras que não haviam sido mencionadas por se terem tornado mais relevantes apenas recentemente. Em ambos os casos é feita uma análise crítica das aplicações, mencionando aspectos importantes como prós, contras e o preço, sendo que a análise é mais profunda no caso das 14 aplicações.

Após uma análise das aplicações o que se notou foi que existe um mercado relativamente segmentado, que pode ser dividido em categorias hierarquizadas. A figura 5 demonstra essa divisão em três categorias distintas.

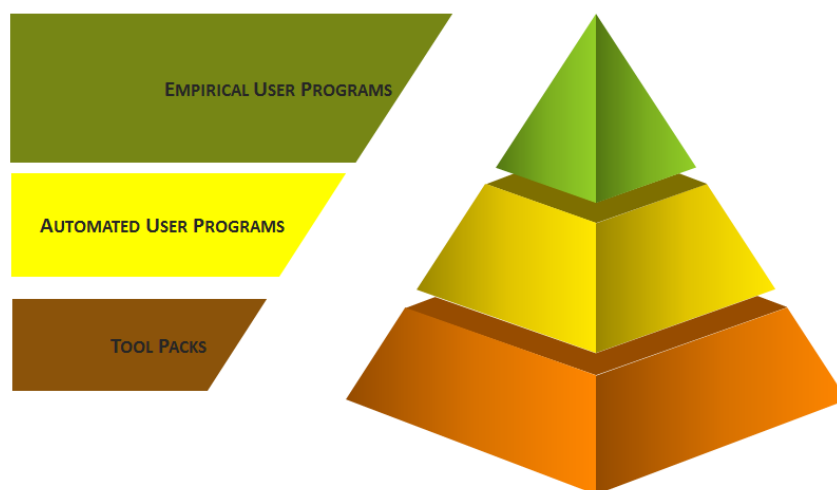


Figura 5– Hierarquização de Aplicações de avaliação de Usabilidade

Na base da hierarquia estão as aplicações denominadas de “*Tool Packs*”. Tal como o nome indica, consistem em vários pacotes de ferramentas com funcionalidades distintas e

---

<sup>4</sup> <http://www.usefulusability.com/24-usability-testing-tools/> (Acedido em 1-08-2017)

<http://www.usefulusability.com/14-usability-testing-tools-matrix-and-comprehensive-reviews/> (Acedido em 1-08-2017)

relacionadas com tarefas necessárias à criação e teste de visualização como *software* de gravação de vídeo ou *click tracers*. Esta categoria forma a base das outras e o seu principal objectivo é o de proporcionar ferramentas suficientemente robustas e diversificadas que permitam às categorias superiores cumprir os seus objectivos, originando assim interfaces de utilizador de melhor qualidade. Um exemplo de *Tool Pack* é o Silverback<sup>5</sup> (Figura 6). O Silverback é uma aplicação que contém vários tipos de ferramentas, apoiando desde a criação e exportação de vídeos até à manipulação e anotações desses mesmos vídeos permitindo realizar algumas experiências controladas ou testes de usabilidade.



Figura 6–Silverback

Em seguida temos a categoria de “*Automated User Programs*”. Esta designação foi atribuída a aplicações capazes de fornecer *feedback*, mas que não recorrem a utilizadores reais, pelo menos não de forma directa. Fazem uso extensivo de *Tools Packs*, situados na camada anterior, aplicando-os numa grande variedade de cenários. O principal objectivo destas aplicações é o de fornecer *feedback* sem recorrer directamente ao utilizador, tentando utilizar outras ferramentas, como por exemplo ferramentas de *logging*, onde os comportamentos dos utilizadores são capturados e analisados, ou em casos mais extremos, fazer uso de *software* inteligente e capaz de simular utilizadores baseados em algoritmos complexos construídos através da análise de modelos de comportamento humano, criando assim um “utilizador sintético”. Em geral estas aplicações são mais económicas e rápidas do que recorrer a

---

<sup>5</sup> <http://silverbackapp.com/> (Acedido em 1-08-2017)

utilizadores reais e são, portanto, consideradas alternativas viáveis aos mesmos fornecendo resultados relativamente aceitáveis ainda que mais limitados. A título de exemplo temos a aplicação Feng-GUI<sup>6</sup>, (Figura 7). Esta utiliza inteligência artificial para fazer uma simulação de “olhos humanos” sobre uma interface ou *website*, fornecendo posteriormente *feedback* na forma de *highlights* e *heat maps*.



Figura 7–Feng- GUI

Finalmente, chegamos à categoria “*Empirical User Programs*”. Estes são por norma serviços oferecidos através de *websites* e, tal como os “*Automated User Programs*”, têm como principal objectivo o fornecimento de *feedback*. No entanto, os “*Empirical User Programs*” recorrem a utilizadores reais através de *software* próprio. Em quase todos os casos as aplicações desta categoria tomam a forma de bases de utilizadores que realizam diversos objectivos, como realização de tarefas específicas ou análise de interfaces, proporcionando depois *feedback* que pode tomar vários formatos desde questionários até vídeos das suas reacções aquando do cumprimento do objectivo proposto. As vantagens deste tipo de aproximação são a grande variedade de respostas e opiniões que é possível obter, proporcionando assim um *feedback* extremamente rico. Existem algumas desvantagens na sua utilização, nomeadamente o facto dos custos e do tempo despendido serem superiores aos das alternativas que não fazem uso de utilizadores reais, existindo também limitações ao nível do controlo sobre que tipo de utilizador irá realizar a tarefa, pelo que pode ser recebido *feedback* de um utilizador que não é representativo do público-alvo da aplicação. Existe também o problema de análise do *feedback* recebido, não sendo muitas vezes perceptível o porquê das reacções dos utilizadores. Um excelente exemplo de aplicação desta categoria é o Try My UI<sup>7</sup>(Figura 8). Este *website* fornece utilizadores para uma grande variedade de testes tendo, no entanto, uma política relativamente diferente dos restantes onde cada utilizador deve criar um vídeo a documentar

---

<sup>6</sup> <http://www.feng-gui.com/> (Acedido em 1-08-2017)

<sup>7</sup> <https://www.trymyui.com/> (Acedido em 1-08-2017)



a sua experiência de utilização, permitindo assim uma avaliação mais personalizada e eliminando algumas incertezas inerentes à avaliação com recurso a utilizadores.

## Remote user testing



Figura 8–Try My UI

Observa-se assim a existência de categorias bem definidas, não tendo sido possível, no entanto, encontrar qualquer exemplo de aplicações focadas em Visualização, reforçando a ideia de que existe uma possível necessidade de aplicações deste género, especialmente quando desenvolvidas com especial atenção às necessidades específicas dos utilizadores que as podem vir a usar.

## 2.4 – Desenvolvimento Centrado no Utilizador e Personas

O *design* de uma aplicação, desde as fases iniciais, é sempre um ponto crítico que será determinante no seu sucesso. Este *design* é ainda mais importante para uma aplicação que pretende dar resposta a necessidades muito específicas como é o caso da que está a ser desenvolvida.

Ao longo deste trabalho foi seguida uma filosofia centrada no utilizador que, quando associada ao uso de Personas, foi essencial para o desenvolvimento de um trabalho mais ligado ao utilizador de forma a melhor se adequar às necessidades às quais se pretendia dar resposta.

## 2.4.1 – Desenvolvimento Centrado no Utilizador

Uma filosofia baseada em *User-centered design (UCD)* foi considerada a melhor aproximação tendo em vista o sucesso da aplicação. A UCD ou *user-driven development (UDD)* é uma *Framework* de processos na qual é dada extrema atenção a aspectos como usabilidade, características do utilizador, ambiente de utilização, *workflow* e tarefas a realizar em cada fase do processo de desenvolvimento, colocando o utilizador como principal motor da definição de todos estes aspectos.

Segundo Mayhem (1999), existem três fases distintas em UCD: *Requirements Analysis*, *Design/Testing/Development* e *Installation*.

A fase de *Requirements Analysis* consiste na criação de um perfil de utilizador com características relevantes para o *design* da interface de utilizador. Esta criação permite decisões de interface centradas no utilizador, bem como a identificação de categorias do mesmo. Consiste também na análise contextual de tarefas onde são estudados extensivamente o *workflow*, tarefas e ambientes de utilização, resultando numa melhor percepção e especificação do utilizador final que serão utilizados para melhor definir objectivos de usabilidade e melhor *design* de interface, sendo também nesta fase feitas *guidelines* gerais de *design* e analisada em detalhe a plataforma de desenvolvimento.

A segunda fase começa por um *redesign* das tarefas com base na análise de requisitos da primeira fase com vista a uma melhoria no *workflow* do utilizador. Nesta fase são também criados os modelos conceptuais em formato de *mockup* ou protótipos que devem incorporar as novas tarefas, não devendo existir um *focus* num *design* completo e funcional. Estes protótipos são depois sujeitos a uma avaliação com técnicas de avaliação iterativas, tais como testes formais de usabilidade. Estes testes serão constantes mesmo durante o desenvolvimento da aplicação. Após estes testes, serão melhorados os vários ecrãs da interface tanto a nível de aspecto como de funcionalidade. Segue-se então o desenvolvimento da aplicação, tendo o cuidado de seguir as *guidelines* estabelecidas e de ter em conta todo o *feedback* adquirido durante as fases anteriores.

A última fase, *Installation*, contempla o intervalo de tempo após a instalação do produto, em que é recolhido *feedback* dos utilizadores para alimentar uma melhoria no *design* actual e no de futuros produtos relacionados ou não com o actual.

## 2.4.2 – Personas

A definição de público-alvo e requisitos de uma aplicação raramente é um processo simples, existindo um problema muitas vezes ignorado: a desumanização do utilizador. Tal como o nome indica, os utilizadores passam a ser somente agrupados em categorias de utilizador e desenvolvem-se cegamente os requisitos para esses grupos, ignorando-se a vida quotidiana e os desafios que os mesmos podem enfrentar e, por conseguinte, muitas vezes são ignorados aspectos intimamente relacionados com as aplicações e que devem ser tidos em conta durante o processo de desenvolvimento.

Para a resolução deste problema podem ser consideradas Personas, um instrumento que visa suportar o processo de definição de requisitos através de uma caracterização mais completa dos utilizadores, auxiliando assim no desenvolvimento de requisitos mais adequados aos utilizadores finais.

As personas são modelos descritivos de utilizadores que funcionam como ferramenta de suporte ao desenho de sistemas interactivos (Cooper et al., 2007) e que têm sido usadas no âmbito de abordagens de desenvolvimento centradas no utilizador (Miaskiewicz et al., 2011). Estas, em conjunto com cenários de aplicação, fornecem uma forma precisa de pensar e comunicar sobre como os utilizadores se comportam e pensam, bem como os porquês das suas atitudes. Apesar de não serem pessoas reais, as personas são baseadas em comportamentos e motivações reais que são representativos das mesmas. Um dos objectivos para a utilização de personas é o de conseguir comunicar quais as principais características e motivações do utilizador, em relação ao sistema que se pretende desenvolver, de uma forma mais humana do que uma mera lista de características, e que possa gerar níveis superiores de empatia (Adlin et al., 2010). Para além disso, a literatura mostra que a consideração de personas resulta, de forma consistente, em ideias relevantes para o contexto abordado e pode até aumentar a capacidade de geração de ideias (So et al., 2017).

Apesar de um conceito simples, as personas têm de ser aplicadas com algum cuidado, sendo que estereótipos e generalizações não são formas adequadas da sua utilização. O seu *design* deve também ser completo e profundo para se obter o efeito desejado.

Um excelente exemplo da utilização de personas pode ser visto em (Cooper et al., 2007)(pag. 78) ilustrando o *design* de um automóvel. Para o efeito são criadas três personas representando três utilizadores diferentes com diferentes objectivos na utilização do automóvel. Enquanto uma dessas personas pretende um automóvel seguro e confortável, a outra prefere um automóvel capaz de transportar grande quantidade de carga e que seja fiável. Estes detalhes muitas vezes são ignorados, existindo a tendência de generalizar, afirmando que todos os utilizadores que pretendem um automóvel pretendem as mesmas características. As personas são assim representativas de pessoas individuais mas que podem, no entanto, também ser representativas de grupos de utilizadores ou de casos limite de utilização.

Assim, algumas das vantagens da utilização de personas (Miaskiewicz et al., 2011) são:

- Determinar o que um produto deve ou não fazer e como se deve comportar através dos objectivos indicados pelas motivações das personas criadas.

- Medição na eficiência do *design* devido à facilidade de testar as soluções com várias personas que são representativas de diferentes utilizadores.

- Maior consenso e comprometimento para com o *design* devido a uma maior compreensão pelas várias partes envolvidas devido à linguagem comum proporcionada pelas personas, o que facilita também a comunicação entre os elementos da equipa de projecto e desenvolvimento e os diferentes grupos de utilizadores que estabelecem os requisitos.

- Contribuição para outras partes do desenvolvimento do produto como *marketing* e diferentes planos de monetização devido a uma maior ligação entre as personas e utilizador final.

Com a utilização de personas pretende-se que o sistema final esteja melhor orientado e adequado às necessidades dos utilizadores finais, resultado numa melhor recepção por parte dos mesmos. De realçar que as personas, para além de uma descrição das suas principais características, devem incluir uma motivação que define, de forma sucinta, as suas principais expectativas em relação ao domínio de aplicação que se pretende abordar, e que, no caso deste trabalho, é a avaliação de visualizações considerando métodos analíticos.

## 3 – Análise de Requisitos e Modelo Conceptual

Um dos aspectos importantes no desenho e desenvolvimento de aplicações é a definição dos requisitos que, por sua vez, dependem das características e motivações dos utilizadores. Este capítulo apresenta o trabalho realizado para definição do público-alvo, através de personas e o estabelecimento de cenários de utilização de onde se derivam os requisitos iniciais da aplicação. Adicionalmente, é também apresentado o resultado de estudos iniciais, com base em protótipos de baixa fidelidade, para definição do modelo conceptual a adoptar para a aplicação.

### 3.1 – Definição do público-alvo

Quando estamos a falar de uma aplicação para um público mais restrito (e.g., devido aos seus interesses, contexto ou dificuldades), torna-se importante fazer uma correcta caracterização dessas especificidades de modo a servir convenientemente as suas motivações e objectivos.

No início desta dissertação foi proposto que o público-alvo deveria ser investigadores e docentes com experiência na área pois seriam os principais interessados e, munidos de conhecimentos da área, seriam os mais adequados para a utilização do *software*. No entanto, à medida que o projecto se foi desenvolvendo, foi vista a necessidade de apoiar um público-alvo mais abrangente.

Foi assim feito um novo levantamento de possíveis requisitos e utilizadores que poderiam beneficiar da utilização desta aplicação. Esse levantamento e conclusões são apresentados na figura 9.

Alunos	Investigadores	Empresas
<ul style="list-style-type: none"> <li>• Opções e Templates</li> <li>• Tooltips explicativos e Feedback</li> </ul>	<ul style="list-style-type: none"> <li>• Interface intuitiva e flexível</li> <li>• Fácil Partilha</li> <li>• Opções e Templates</li> </ul>	<ul style="list-style-type: none"> <li>• Fácil Partilha</li> <li>• Interface intuitiva e fácil de integrar</li> <li>• Opções e Templates</li> </ul>

**Figura 9 - Levantamento de Público-alvo e Requisitos**

Conforme pode ser observado, foi considerado que alunos e equipas de desenvolvimento em empresas de produção de *software* poderiam também beneficiar com a aplicação. Foi também observado que muitos investigadores eram também docentes pelo que essa parcela de utilizadores está agrupada junto com os investigadores.

O acréscimo dos alunos foi feito a pensar numa utilização mais didáctica deste *software*. Em ambiente exploratório ou em sala de aula, o mesmo seria utilizado para fins educacionais e aprendizagem através de uma aproximação “*hands-on*”. Para isso seriam necessárias opções relacionadas com fornecimento de *templates*, heurísticas, ferramentas de criação de interfaces bem como um manual e *tooltips* explicativas de modo a reduzir a curva de aprendizagem.

Os investigadores e docentes continuam a ser um público importante necessitando, tal como os alunos, de flexibilidade nas opções e ferramentas de interface disponíveis de forma a poderem experimentar na criação de novas visualizações. Continuam ainda a ser necessários mecanismos de obtenção de *feedback* como questionários, bem como opções para a sua partilha em conjunto com as visualizações criadas.

Finalmente foram acrescentadas as equipas de produção de *software* nas empresas como público-alvo a captar, principalmente após a análise da ferramenta “*Quant-Ux*”. Desenvolver a pensar em apelar a um domínio empresarial é, no entanto, bastante mais complexo devido não só à resistência à mudança existente no meio, mas também devido a uma maior exigência. Alguns dos principais requisitos a destacar são a capacidade de partilha de protótipos entre as equipas de desenvolvimento (*testers*, programadores, etc.), interface intuitiva e de fácil integração no ambiente de desenvolvimento actualmente em vigor, bem como opções para a criação de interfaces e visualizações relativamente

robustas e variadas para que não exista um grande limite à sua criação por parte das equipas de *design*.

Apesar da definição das empresas como público-alvo, devido à fase inicial em que se encontra o projecto, foi decidido que o *focus* inicial seria nos alunos e investigadores/professores, pelo que estes foram analisados neste dissertação de forma mais profunda utilizando uma ferramenta específica para extracção de requisitos: as personas.

## 3.2 – Personas Criadas

Durante a concepção deste trabalho foram estudados vários grupos de utilizadores passíveis de serem o público-alvo. Conforme discutido anteriormente os principais grupos encontrados foram alunos, investigadores/professores e *developers* em empresas. Após uma discussão dos resultados obtidos foi decidido que o desenvolvimento da aplicação iria focar-se nos professores e alunos, sendo que os professores poderiam também ser investigadores na área da Visualização.

Assim foram criadas duas personas, o Carlos e a Isabel, com o intuito de representarem um aluno e uma investigadora respectivamente. O principal objectivo na criação destas personas foi uma melhor definição, compreensão e comunicação das características do público-alvo com o intuito de identificar os requisitos que vão reger o desenvolvimento do sistema pretendido. Tendo em conta o público-alvo, a definição das características das personas foi efectuada com base no conhecimento existente na equipa, composta pelo autor e orientadores, com experiência enquanto alunos/professores nas áreas de Interfaces Humano-Computador e Visualização de Dados e Informação. Esta abordagem procurou simplificar a criação das personas (dispensando um estudo mais aprofundado com base na literatura e entrevistas a potenciais utilizadores, por exemplo (Cooper et al., 2007)), tendo em conta os constrangimentos temporais para a realização deste trabalho, mas mantendo, ainda assim, perfis julgados representativos dos utilizadores e suas motivações. As personas apresentadas são já o resultado de um conjunto de iterações de análise e discussão das suas características e motivações.

O Carlos representa um aluno, um dos segmentos que se pretende que utilize a aplicação. Neste caso os alunos são alunos de informática, com alguma experiência relativamente à área de Visualização ou a iniciarem a sua formação na mesma, pelo que isso deve ser reflectido na persona. Os principais aspectos a ter em conta durante a construção da sua persona foram: caracterização biográfica, formação, competências, interesses e motivação.

A persona Carlos<sup>8</sup> é descrita da seguinte forma:



**“Carlos Rodrigues**, nascido a 8 de Abril de 1995 em Ílhavo no distrito de Aveiro, Portugal, vive com o seu pai e a sua mãe e é filho único. Actualmente frequenta o segundo ano do curso de Engenharia Informática da Universidade de Aveiro e neste contexto frequenta a disciplina de Interação Humano-Computador (IHC) onde tem estudado diferentes métodos empíricos e analíticos de avaliação de usabilidade.

O Carlos seguiu um percurso escolar relativamente normal, tendo integrado a universidade de Aveiro após ter estudado na área de Ciências e Tecnologias tendo por isso fortes bases em língua Inglesa e Matemática. O Carlos domina vários aspectos da informática, sabendo utilizar todas as funcionalidade básicas de um computador e estando confortável com funcionalidades mais avançadas.

No contexto escolar, é um aluno aplicado, sendo bastante curioso especialmente no que toca a novos desenvolvimentos nas áreas cobertas pelo seu curso e gosta de explorar as matérias, sempre que possível, de forma autónoma.

**Motivação:** O Carlos quer conseguir perceber melhor os diferentes aspectos de uma avaliação suportada em métodos analíticos e ter uma forma mais sistemática de fazer as avaliações e agregar os resultados.”

Por outro lado, a Isabel, representa o segmento de investigadores. Pertencendo a um segmento bastante educado em Visualização de Informação é importante que esta característica seja reflectida na sua persona. Os principais aspectos que foram tidos em conta durante a sua criação foram: caracterização biográfica, formação, caracterização da carreira actual (docência e investigação), domínio da tecnologia, interesses específicos enquanto docente/investigadora e a sua motivação.

---

<sup>8</sup> Imagem ilustrativa da Persona descrita retirada de <https://commons.wikimedia.org/wiki/File:Sajal.jpg> (Acedido em 1-08-2017)



A persona da Professora Isabel<sup>9</sup> é descrita da seguinte forma:



“**Isabel Custódio**, nascida a 18 de Março de 1984 na Cruz de Pau, no distrito de Setúbal, Portugal, vive actualmente com o seu marido e dois filhos. É investigadora na Universidade de Aveiro, leccionando também a disciplina de Interação Humano-Computador (IHC) do curso de Engenharia Informática.

Ao longo do seu percurso profissional tem leccionado e investigado nos domínios da Usabilidade de Sistemas Interactivos e da Visualização de Dados e Informação. Nesse âmbito, o seu interesse tem-se focado na proposta e teste de métodos analíticos de avaliação de usabilidade como forma de guiar o desenvolvimento de visualizações.

Quer nas aulas, quer na sua investigação, sente que as avaliações deveriam poder ser feitas de modo a que os dados resultantes (por exemplo, heurísticas violadas) pudessem mais facilmente ser partilhados e comparados. Isto facilitaria uma aplicação mais sistemática dos métodos abordados, por parte dos alunos, e uma mais fácil recolha de dados para investigação.

**Motivação:** A professora Isabel gostaria de ter uma forma mais estruturada e fácil de propor tarefas de avaliação aos seus alunos, em contexto de aula, e de suportar o seu trabalho de investigação sobre avaliação no contexto de sistemas de visualização de dados e informação.”

Ficam assim definidas e caracterizadas as personas que irão ser utilizadas no passo seguinte desta dissertação: a definição de cenários e requisitos.

Com as personas definidas, é agora possível passar à fase de extracção de requisitos. Para uma utilização correta e eficiente das personas, foram criados vários cenários de utilização da aplicação baseados em situações reais e nos quais as personas participam. A vantagem da descrição de cenários é a de que permite definir, de forma mais explícita, o conjunto de acções que o utilizador deve poder efectuar.

Para cada persona foram definidos três cenários distintos e, de cada cenário, foram extraídos os requisitos correspondentes que são depois compilados numa lista de requisitos finais para o sistema a desenvolver. Os cenários apresentados são o resultado de um processo iterativo de discussão que, partindo de uma versão inicial, procurou focar-se nos aspectos considerados mais importantes para o auxílio à avaliação de visualizações.

---

<sup>9</sup> Imagem ilustrativa da Persona descrita retirada de [https://commons.wikimedia.org/wiki/File:12\\_01\\_2016\\_Alice\\_headshot.jpg](https://commons.wikimedia.org/wiki/File:12_01_2016_Alice_headshot.jpg) (Acedido em 1-08-2017)

Assim os cenários para o Carlos são os seguintes:

**Cenário 1:**

O Carlos está a ter o primeiro contacto com a técnica de avaliação heurística, nas aulas de IHC. Para que os alunos percebam melhor no que consiste uma avaliação heurística, a Professora disponibilizou alguns exemplos de avaliações heurísticas feitas em anos anteriores. O Carlos abre os exemplos na aplicação e pode ver imagens, com diferentes violações de heurísticas assinaladas e alguns comentários explicativos.

**Requisitos Cenário 1:**

- Possibilidade de *upload* e visualização de exemplos de avaliações
- Possibilidade de assinalar potenciais problemas de usabilidade classificados de acordo com as heurísticas que são violadas e respectivos graus de severidade.

**Cenário 2:**

O Carlos tem de fazer uma avaliação heurística de um *site* para um dos trabalhos de IHC. Abre a aplicação e inicia uma sessão de avaliação incluindo as heurísticas de Nielsen. De seguida abre o *site* no *browser* e começa a análise. Ao detectar o primeiro problema potencial, faz uma captura de ecrã, do *site* e, na aplicação, inclui essa captura. De seguida assinala, sobre a imagem, os aspectos que apresentam problemas e a(s) respectiva(s) heurísticas violadas bem como o grau de severidade do problema. Como ainda não está muito familiarizado com as heurísticas, recorre aos textos explicativos, para verificar se a heurística escolhida pode ter a interpretação que lhe está a dar. Volta ao *browser* e repete este procedimento até ter analisado todo o *site*. No final, ao terminar a avaliação, salva o trabalho e escolhe a geração de um PDF com o relatório.

**Requisitos Cenário 2:**

- Inclusão de alguma forma de captura de ecrã onde possam ser assinalados os potenciais problemas de usabilidade identificados
- Várias heurísticas com explicação e graus de severidade
- Geração de um relatório em formato PDF.

**Cenário 3:**

O Carlos, depois de ter frequentado a cadeira de IHC, está agora a frequentar Visualização de Dados e Informação. No contexto de um dos trabalhos, pretende comparar ferramentas de visualização de dados geológicos, para apresentar aos colegas. Para não se restringir a uma comparação em termos de funcionalidades, quer fazer uma avaliação da usabilidade. Decide usar a ferramenta de avaliação usada em IHC. Ao criar uma sessão de avaliação, verifica que existe um conjunto de heurísticas específicas para visualizações e decide adoptá-las. A aplicação disponibiliza-lhe notas explicativas para cada uma das heurísticas que ele estuda antes de fazer as avaliações.

**Requisitos Cenário 3:**

- Existência de vários conjuntos de heurísticas para avaliações específicas e notas explicativas para as mesmas.

Seguidamente são também apresentados os cenários da professora Isabel:

**Cenário 1:**

A Professora Isabel pretende que os seus alunos utilizem um novo conjunto de heurísticas para avaliar uma aplicação. Fornece aos alunos uma nova lista. Ao iniciar a aplicação, a nova lista de heurísticas já se encontra disponível para utilização.

**Requisitos Cenário 1:**

- Permitir adicionar listas de heurísticas personalizadas.

**Cenário 2:**

A Professora Isabel pretende obter a colaboração de alguns colegas para perceber se o protótipo de uma aplicação de visualização que a sua equipa está a desenvolver tem problemas de usabilidade. Para tal, abre uma sessão de avaliação, estabelecendo o uso de um conjunto de heurísticas proposto por si, e onde coloca as diferentes vistas da interface, de modo a que os colegas não tenham de o fazer. Seguidamente, escolhe enviar a sessão criada aos colegas, em modo de avaliação, que limita a alteração de alguns aspectos da sessão e basicamente permite contributos sobre a estrutura que ela delineou.

**Requisitos Cenário 2:**

- Sessões de avaliação com limitação de edição e conjunto específico de heurísticas a considerar
- Partilha de sessões entre pares.

### Cenário 3:

A Professora Isabel está a trabalhar sobre um novo conjunto de dados e pretende perceber qual a melhor visualização para responder a um conjunto de questões que lhe foram fornecidas como sendo as mais usuais, no dia-a-dia dos técnicos que analisam esses dados. Utilizando a aplicação, cria uma sessão de avaliação constituída por uma série de questões, algumas de resposta livre, outras com várias opções, acompanhadas por imagens contemplando várias visualizações e pontos de vista. As respostas deverão permitir perceber qual a melhor visualização para responder mais facilmente a cada questão.

#### Requisitos Cenário 3:

- Criação de experiências controladas/Questionários e recepção de *feedback* sobre os mesmos.

Após uma análise de todos os cenários e os requisitos extraídos de cada um deles, foi criada uma lista de requisitos iniciais que a aplicação deve suportar:

- Criar sessões de avaliação
- Disponibilizar diferentes listas de heurísticas e sua explicação
- Permitir adicionar novas listas de heurísticas
- Adicionar capturas de ecrã às sessões de avaliação
- Permitir assinalar, sobre as imagens, as heurísticas violadas
- Permitir associar a cada problema uma severidade
- Permitir adicionar comentários livres aos problemas assinalados ou a aspectos considerados relevantes da interface
- Salvar sessões de avaliação
- Permitir partilhar sessões de avaliação
- Permitir criar “*templates*” para servirem de base a avaliações feitas por outras pessoas. Por exemplo, definindo, à partida as capturas de ecrã a considerar e a lista de heurísticas
- Permitir criar “*templates*” de avaliação contemplando questionários
- Gerar um relatório a partir dos dados contidos numa sessão de avaliação.

Estes foram os requisitos que guiaram todo o processo de desenvolvimento da aplicação e que foram obtidos com recurso à utilização de personas e cenários que torna mais explícita a sua proveniência.

### 3.3 – Modelo conceptual da aplicação

Tal como foi dito anteriormente, toda a aplicação seguiu uma filosofia de *user-centered design*. O resultado foi assim uma aplicação que tem como objectivo primário ser flexível e intuitiva seguindo um fluxo de trabalho e uma separação das duas componentes de avaliação suportadas num pequeno número de interfaces de modo a reduzir a curva de aprendizagem.

Logo na janela inicial da aplicação é possível observar esta separação sendo que imediatamente é dada a escolher a opção de prosseguir para a interface de experiências controladas ou para Avaliações Heurísticas. Desta forma foi possível desenvolver interfaces específicas para cada método de avaliação.

Seguidamente e independentemente do método de avaliação seleccionado, existe um menu de projecto. O objectivo deste menu é permitir ao utilizador algum nível de controlo sobre a organização das suas avaliações. Por exemplo, o utilizador pode criar um projecto que seria uma avaliação para determinada empresa e posteriormente qualquer avaliação para essa empresa seria associada a esse projecto, dando assim alguma flexibilidade extra à aplicação.

Finalmente, a interface principal foi pensada tendo em conta o fluxo de trabalho que o utilizador segue, sendo que existe uma área central onde a imagem é carregada e onde todas as funcionalidades existentes para manipular e trabalhar a imagem se encontram em menus laterais ou numa *toolbar*. O que isto permite é um claro foco na imagem ao mesmo tempo que todas as funcionalidades são facilmente acessíveis, tornando assim a interface mais intuitiva e aliviando a curva de aprendizagem da mesma.

### 3.4 – Evolução do protótipo

Aquando do começo da dissertação, existiam vários requisitos necessários para a criação desta aplicação, sem conhecimento concreto e comprovado do público-alvo. Ao longo do processo de desenvolvimento e antes ainda da definição desse público ou formalização dos requisitos finais, existiram vários protótipos conceptuais, cada um correspondente a uma evolução no *design*, contendo novas características e requisitos. Nesta secção são apresentadas as duas versões conceptuais mais importantes, correspondendo ao início da dissertação e uma versão mais avançada, bem como as suas características pois considera-se importante dar uma noção da evolução do projecto até à versão final.

### 3.4.1 – Primeira Versão Conceptual

A primeira versão conceptual foi o ponto inicial da dissertação, tendo sido resultado de várias sessões de “*brainstorming*” com os orientadores. A aplicação teria a forma de aplicação *web* de apoio a avaliação de visualizações, nomeadamente experiências controladas.

As experiências controladas são um dos métodos usados em avaliação de visualização onde existem necessidades especiais, nomeadamente nas suas características para que possa ser replicada por várias pessoas e a obtenção de resultados se mantenha consistente. O *feedback* obtido e a sua partilha é assim algo extremamente importante para que se possa gerar discussão sobre os resultados das experiências controladas. Todas estas considerações revelaram um claro foco em experiências controladas durante esta primeira versão.

Para atender a esta realidade, eram necessárias ferramentas para a criação dessas mesmas experiências, ferramentas essas que teriam de incluir a capacidade de elaborar questionários, incluindo imagens, listas de heurísticas, etiquetas temporais das respostas, entre outras funcionalidades relevantes. Outra importante preocupação seria a partilha dessas experiências, sendo que a forma teria de ser fácil tendo sido pensado no formato de *link* para essa partilha, que facilmente poderia ser enviado a um investigador para que observasse a experiência e pudesse replicá-la. Finalmente seria necessária uma forma de obter *feedback* dos utilizadores após o preenchimento dos questionários, sendo que este é sempre um dos pontos mais críticos das visualizações. Uma possível solução encontrada seria a inclusão de uma mensagem do utilizador para o investigador no final do questionário com qualquer comentário ou *feedback* relevante.

Foram estas as principais funcionalidades e requisitos pensados para a primeira versão, que apesar de relativamente básica serviu como uma base sólida sobre a qual foi possível evoluir.

### 3.4.2 – Segunda Versão Conceptual

A segunda versão foi fruto de uma pesquisa mais intensiva, com especial menção para a análise do Quant-Ux, que serviu de inspiração para a mesma. Esta segunda versão pretendia mudar um pouco a forma como a aplicação iria funcionar, dando-lhe uma maior flexibilidade, mais funcionalidades e pretendendo apoiar um público-alvo mais diverso, não estando tão focada na questão das experiências controladas.

Este novo conceito trouxe também uma crescente importância das avaliações heurísticas, tendo sido com o objectivo de melhor as suportar que se pretendeu adoptar um mecanismo de *screen capture*, na forma de vídeo. Esse vídeo podia depois ser manipulado e retirados *snap shots* onde poderiam depois existir anotações de problemas de usabilidade relacionados com

violações heurísticas. Queria-se assim criar uma aplicação mais flexível e abrangente e que continuasse a especializar-se no apoio a avaliação de visualizações.

Esta segunda versão foi assim uma importante evolução em relação à primeira, inspirando um novo fluxo de trabalho onde a existência de um vídeo ou imagem fornecida pelo utilizador passa a ser o componente central, o que leva também à existência de funcionalidades que suportem esta nova forma de trabalhar. Esta aproximação e características estão ainda presentes na versão mais recente da aplicação.

### 3.4.3 – Versão Actual do Protótipo Funcional

A versão actual da aplicação é bastante semelhante à segunda versão conceptual. Tal como ela, prevê uma separação das experiências controladas e das avaliações heurísticas com uma interface específica para cada uma delas. Continua a existir um fluxo de trabalho relativamente simples e focado numa imagem central à qual estão associadas várias funcionalidades e atribuição de heurísticas.

Existe, no entanto, um esboço de um sistema de gestão de projectos. Este sistema permite agrupar visualizações em vários projectos e assim criar um sistema de organização para o utilizador. O sistema está apenas em fase de *design*, não estando implementado aquando da entrega desta dissertação.

A maior diferença reside numa clara mudança da plataforma de desenvolvimento. Até esta versão a principal plataforma para a aplicação sempre foi no formato web. Apesar de ainda ser uma componente importante, foi decidido que a aplicação seria desenvolvida para *desktop* primeiro para evitar algumas das complicações específicas de desenvolver para web, especialmente ao nível da interface.

## 3.5 – Evolução e Prototipagem das Interfaces de Utilizador

Na secção anterior, o resultado final da utilização de Personas foi um conjunto de requisitos que devem estar presentes na versão final do programa. Estes requisitos serão as *guidelines* durante o desenvolvimento mas devem ser implementados cuidadosamente através de uma conceptualização e, idealmente, através de testes com utilizadores.

Para ajudar na concepção foi utilizado um protótipo de baixa fidelidade, com base no qual foram realizados testes com utilizadores a fim de obter *feedback* para orientar o desenvolvimento de um protótipo que mais tarde seria novamente testado.

### 3.5.1 – Protótipo de Baixa Fidelidade

Conforme mencionado anteriormente, para esta fase inicial de obtenção de *feedback* de utilizadores foi utilizado um protótipo de baixa fidelidade (Rudd et al., 1996), nomeadamente um protótipo realizado com recurso ao Balsamiq Mockups 3<sup>10</sup> que tem todas as características da prototipagem em papel sendo, no fundo, um “protótipo em papel” digital e interactivo.

Este método de prototipagem<sup>11</sup> é um método rápido e barato que permite testar ideias antes de desenvolver um protótipo funcional tendo inúmeras vantagens durante a sua criação como, por exemplo, o facto de permitir rápidas alterações que podem ser efectuadas mesmo durante o processo de testes com os utilizadores. Mesmo quando aplicado digitalmente é mais rápido que outros métodos, permitindo assim envolver o utilizador mais cedo (Fonseca et al., 2012).

É também relevante afirmar que se estima que até 80% dos problemas de usabilidade podem ser detectados com recurso a prototipagem em papel e existe uma tendência para os utilizadores serem mais críticos com este tipo de prototipagem do que com uma aplicação formal e funcional.

Protótipos deste género são especialmente adequados para obter *feedback* no que respeita a:

- Conceitos e terminologia;
- Navegação;
- Conteúdo formal;
- Funcionalidade.

Com recurso a este protótipo foi feito um teste de usabilidade para validar algumas das implementações dos requisitos que tinham sido definidos.

---

<sup>10</sup> <https://balsamiq.com/products/mockups/> (Acedido em 1-08-2017)

<sup>11</sup> <https://www.smashingmagazine.com/2014/10/the-skeptics-guide-to-low-fidelity-prototyping/> (Acedido em 1-08-2017)



### **3.5.2 – Teste de Usabilidade do protótipo**

Tal como outros testes de usabilidade, também para a prototipagem em papel existem regras que devem ser seguidas para maximizar a eficácia dos testes realizados (Fonseca et al., 2012).

Assim a preparação dum teste com utilizadores baseado num protótipo em papel deve seguir os seguintes passos:

- Definir as tarefas que o utilizador deve realizar;
- Preparar os desenhos, ecrãs, menus, etc... necessários à realização dessas tarefas;
- O teste é realizado com os utilizadores e o papel da reacção da aplicação às várias acções do utilizador é desempenhado por quem desenvolveu o teste, trocando interfaces quando necessário;
- Outro elemento deve observar a experiência e ir tomando notas. Caso seja realizado por uma só pessoa, esta deve tomar notas imediatamente a seguir ao teste;
- O elemento que conduz a experiência não deve dar explicações sobre o funcionamento da interface, mas apenas simular o comportamento da mesma para poder identificar problemas.
- É possível e aconselhado, realizar ao longo dos testes alterações nos protótipos para testar soluções e melhorar a interface.

Estas foram as linhas orientadoras para o teste com utilizadores realizado utilizando o protótipo em Balsamiq, tendo em conta que existia apenas um elemento a conduzir os testes.

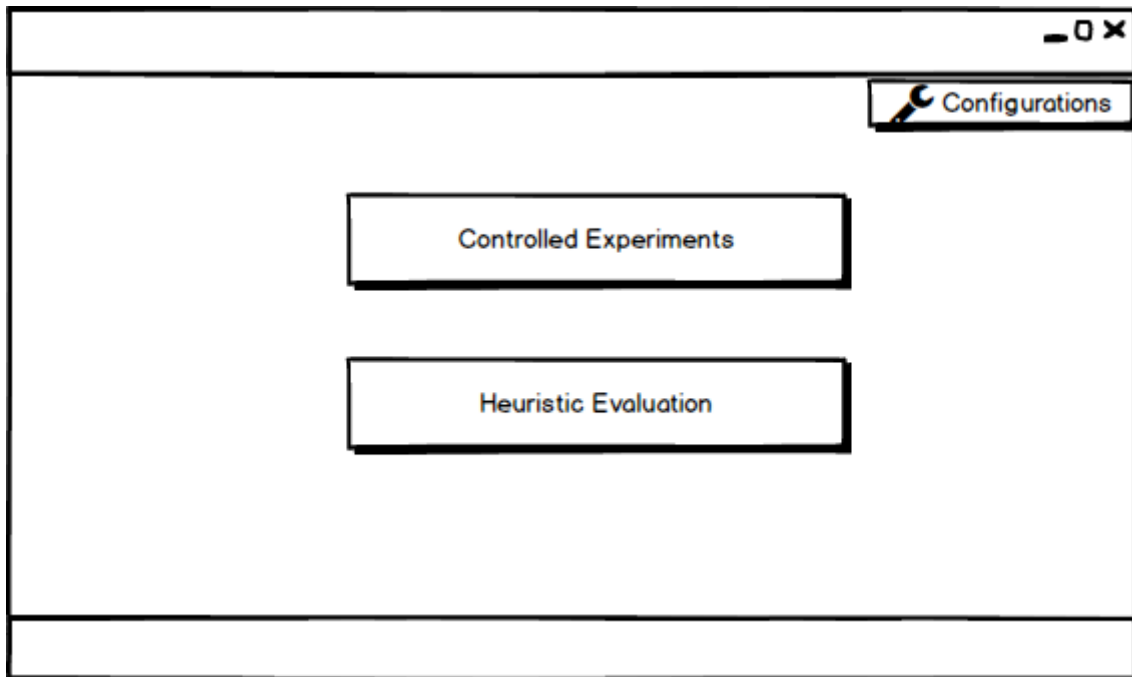
#### **3.5.2.1 – Protótipo**

Para o propósito deste teste foram criadas várias interfaces para que os utilizadores pudessem explorar o conceito inicial da aplicação já com algumas interacções programadas devido à utilização de Balsamiq.

Neste teste foram apenas testadas as interfaces de utilizador referentes às avaliações heurísticas já que o seu desenvolvimento foi estabelecido como prioritário.

Ao longo das próximas páginas serão mostrados aspectos da interface de utilizador, bem como descritos os seus elementos e propósito.

A figura 10 mostra o primeiro menu que o utilizador vê quando corre a aplicação; este menu permite ao utilizador escolher o tipo de método de avaliação que pretende usar.



**Figura 10 - Menu Inicial da Aplicação**

Tal como inicialmente foi delineado, é necessário dividir a aplicação entre experiências controladas e avaliações heurísticas existindo interfaces diferentes, especificamente desenhadas para cada uma destas situações

Existe também um botão de configurações que permitiria ao utilizador configurar alguns aspectos da aplicação. Nesta fase esse menu ainda não se encontra disponível, pois ainda não estavam definidos exactamente quais os aspectos que deveriam de ser personalizáveis.

O menu “*Project Selection*”, mostrado na figura 11, tem como principal função a gestão de projectos de avaliações heurísticas. O objectivo será criar vários projectos aos quais estariam depois associadas visualizações, permitindo ao utilizador gerir projectos de forma independente e organizada.

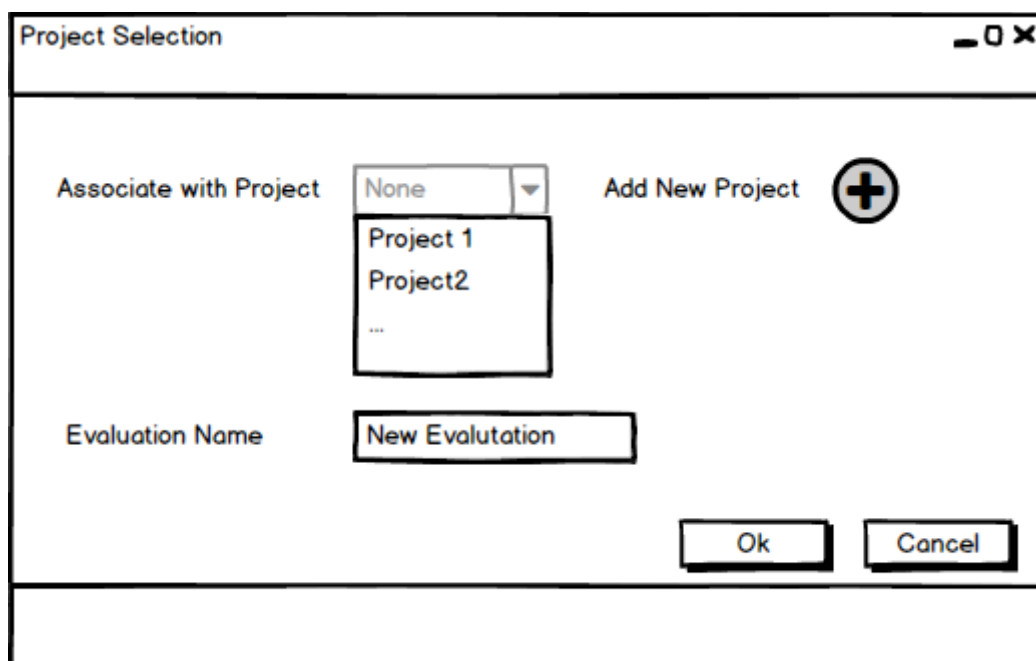


Figura 11 - "Project Selection"

Utilizando o botão "+" o utilizador cria um projecto que será depois disponibilizado na "Combo Box" do "Associate with Project". O utilizador selecciona o projecto criado e a nova avaliação será associada a esse projecto. Quando voltar a inicializar a aplicação, esse projecto estará disponível e poder-lhe-á ser associada outra visualização.

Este sistema foi pensado e desenhado num espaço de tempo relativamente curto para obter *feedback* e verificar qual a aceitação dos utilizadores, pelo que ainda não estava muito desenvolvido aquando da realização deste teste, existindo algumas dúvidas de como será feita a gestão dos projectos.

A figura 12 mostra o principal menu da aplicação onde praticamente todas as funcionalidades são disponibilizadas ao utilizador.

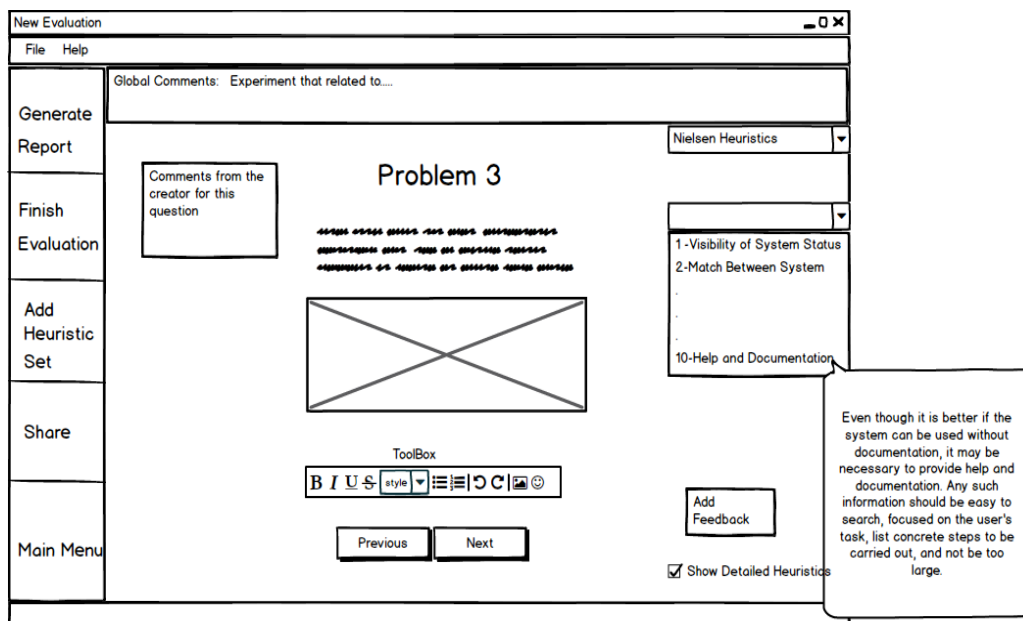


Figura 12 - Interface Principal da Aplicação

A interface está dividida em três partes distintas:

A área central é a área de trabalho onde é mostrada a imagem escolhida pelo utilizador. Começando no topo existe uma área para comentários globais relativos a toda a avaliação e que será persistente em todos os problemas criados. Seguidamente é mostrado o número do problema em que o utilizador se encontra seguido de uma breve descrição, após o qual está uma imagem (representando um aspecto de uma interface ou algo passível de avaliação heurística), que o utilizador deverá fazer *upload* e que será o foco principal da aplicação. Nesta imagem serão associados os potenciais problemas correspondentes a violações de heurísticas, bem como comentários e sinalizações presentes na “Tool Box”. Finalmente, existem botões para navegar entre os vários problemas criados.

Na área à direita são disponibilizadas as funcionalidades relativas aos conjuntos de heurísticas. Em primeiro lugar existe uma “Combo Box” com as heurísticas que estão actualmente a ser visualizadas (neste caso “Nielsen Heuristics”). Quando um conjunto está seleccionado, existirá uma segunda estrutura, aqui representada por uma segunda “Combo Box”, onde estão disponíveis todas as heurísticas correspondentes ao conjunto seleccionado. Estas são as heurísticas que serão colocadas na imagem para indicar o não cumprimento das heurísticas, possivelmente através de “Drag and Drop”. Existem também “tooltips” para cada heurística de modo a orientar melhor o utilizador, um botão para *feedback* do problema em específico (na forma de comentário) e ainda uma opção de ocultar a estrutura que contém as heurísticas de modo a dar mais flexibilidade à aplicação.

Para concluir, no lado esquerdo da interface existe um menu lateral que contém opções relativas à avaliação em curso. A opção “Generate Report” é uma funcionalidade que pretende dar resposta a um requisito que diz que a aplicação deve ajudar o utilizador a elaborar um

relatório de forma automática. Seguidamente a opção “*Finish Evaluation*” dará por concluída a avaliação actual, não permitindo incluir mais problemas. O “*Add Heuristic Set*” permite, conforme o nome indica, adicionar mais “conjuntos de heurísticas”, sendo inclusivamente possível trabalhar com vários em simultâneo. Existe também opção de “*Share*” que abre uma nova janela, permitindo ao utilizador partilhar a avaliação actual. Finalmente o botão “*Main Menu*” irá retornar o utilizador para o menu inicial da aplicação.

A figura 13 mostra uma alternativa à interface de utilizador anterior que é em tudo semelhante à anterior, tendo apenas a pequena modificação da forma dos botões. Foi criada especificamente para ver como os utilizadores reagem à diferença de formato dos botões, pedindo-lhe que expressassem depois a sua preferência.

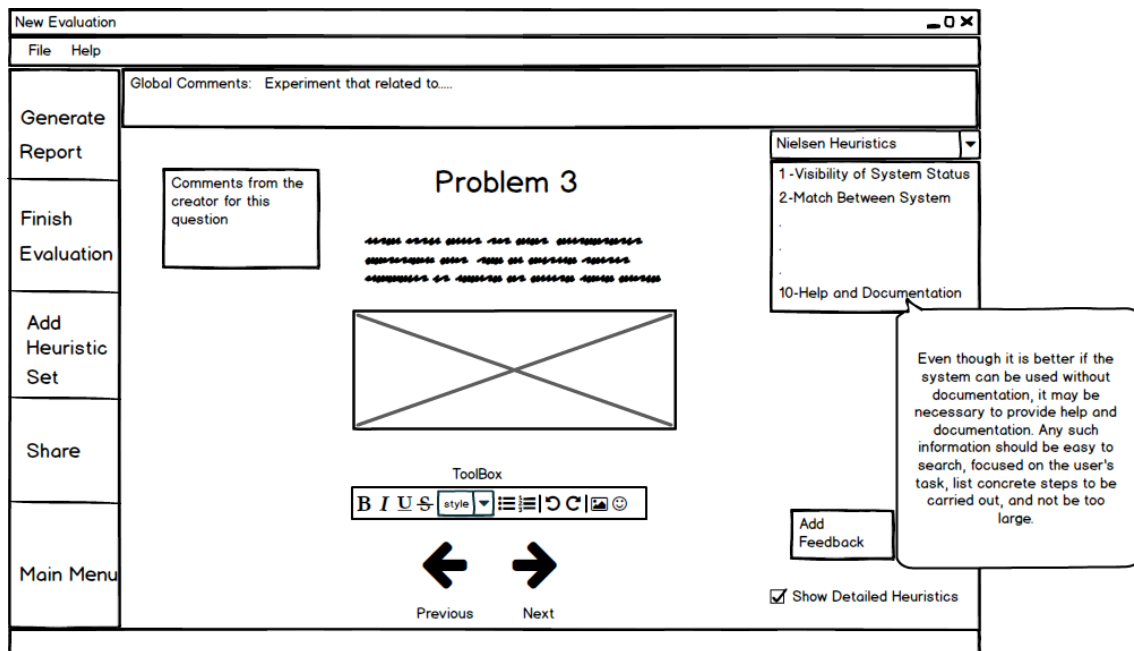


Figura 13 - Interface Alternativa do Menu Principal

A figura 14 mostra a janela *pop-up* que será apresentada quando o utilizador selecciona o botão “Share”.



Figura 14 - Menu de Partilha

Estando planeada uma expansão para uma versão *web*, foi pensado num sistema por “*link*” que pode ser enviado a pessoas com quem o utilizador queira partilhar a avaliação e que poderá ou não ser editável. Esta funcionalidade permite, por exemplo, troca de opiniões sobre a avaliação partilhada.

Nesta fase, a funcionalidade ainda não estava finalizada, estando em curso discussões sobre a melhor forma de fazer um sistema de partilha intuitivo e flexível.

### 3.5.2.2 – Avaliação com base no protótipo

Foi definido um conjunto de tarefas que são inicialmente muito genéricas de forma a detectar problemas mais superficiais, existindo depois algumas mais específicas para permitir obter *feedback* sobre os pontos que mais dúvidas suscitavam.

A seguir apresenta-se o conjunto de tarefas, bem como uma breve explicação do pretendido em cada tarefa e questões colocadas aos utilizadores após a sua realização:

**1 - Considerações Gerais**– No início do teste de usabilidade foi feita uma breve introdução do objectivo da aplicação e depois dado algum tempo ao utilizador para se familiarizar, sendo-lhe depois pedida a opinião de uma forma geral.

**2–Botão “Add Heuristics Set”**– Um dos aspectos que suscitou dúvidas durante o *design* da aplicação foi a forma como seriam adicionados conjuntos de heurísticas à interface principal.

Tendo em vista obter algum *feedback* dos utilizadores, perguntou-se se preferiam a utilização do botão “Add Heuristic Set” ou se seria preferível uma opção na linha final da “Combo Box” de heurísticas.

**3–Formato dos botões de “Next” e “Previous”** – Outro aspecto que suscitou dúvidas foi o formato dos botões utilizados na aplicação. Assim foi-lhes pedida a opinião sobre se o formato actual era o mais adequado ou se preferiam botões com o formato de setas.

**4–Ocultação Opcional das Heurísticas** – A listagem de heurísticas é uma das principais funcionalidades da aplicação e é essencial para a associação das violações à imagem escolhida para caracterizar os potenciais problemas. No entanto, podem ocupar uma porção significativa da interface. Perguntou-se aos utilizadores se achavam que deveria existir uma forma de ocultar esta lista e, em caso afirmativo, se preferiam um menu em formato “combo box” ou um botão (representado por “Show Detailed Heuristics”).

**5–Associação de Heurísticas à imagem por “Drag and Drop” ou dois Cliques** – Como tarefa final os utilizadores expressaram a sua preferência sobre como associar heurísticas da lista à imagem. Foram-lhes apresentadas duas opções, “Drag and Drop” da lista para a imagem ou através de clique na heurística, seguido de clique na imagem.

Foi despendido bastante mais tempo na primeira tarefa do que nas tarefas subsequentes pois era importante perceber como os utilizadores reagiam a toda a dinâmica da aplicação. Entre cada tarefa existiu um pequeno dialogo com os utilizadores não só para melhor compreensão do seu comportamento e escolhas, mas também para registo de *feedback* para posterior análise.

Como comentário final realça-se que este foi uma avaliação ligeiramente diferente de um teste de usabilidade com base num protótipo em papel, existindo um foco maior na obtenção de *feedback* sobre alguns aspectos específicos do funcionamento da aplicação do que na sua funcionalidade global. Esta situação foi motivada não só por algumas restrições de tempo por parte dos utilizadores, mas também por se pretender a maior quantidade de *feedback* possível nesse intervalo de tempo e por já estar agendado um teste com um protótipo mais desenvolvido para uma data posterior onde o foco seria o funcionamento da aplicação como um todo.

### **3.5.2.3 – Feedback e Conclusões**

Esta primeira fase de avaliação foi realizada com a colaboração de quatro alunos de Engenharia de Computadores e Telemática pertencentes a uma turma de IHC (Interacção Humano Computador) da Universidade de Aveiro que se voluntariaram para o efeito e que, devido à sua formação, tinham já alguma experiência em avaliação de interfaces de utilizador.

Cada sessão com um aluno teve uma duração de cerca de vinte minutos com intervalos de cinco minutos para preparação para o aluno seguinte.

Durante as sessões de avaliação os alunos realizaram as tarefas descritas anteriormente e o seu *feedback* foi posteriormente anotado sendo seguidamente apresentadas as conclusões.

No que se refere a considerações gerais, a recepção por parte dos utilizadores foi bastante positiva, tendo estes gostado da interface de forma geral e da separação das componentes de Avaliação Heurística e de Experiências Controladas. A principal preocupação demonstrada foi quanto ao tamanho da área disponibilizada para a imagem, tendo sido argumentado que visto ser o ponto central de todo o processo, a área disponibilizada poderia ser demasiado pequena. Um dos utilizadores demonstrou também alguma resistência ao menu de opções à esquerda, dizendo que preferia que as opções estivessem disponíveis noutro formato, mas sem conseguir sugerir qual poderia ser. Finalmente, um dos alunos questionou quanto à persistência da imagem entre problemas e chamou a atenção para a necessidade de uma forma de navegação mais fácil entre problemas, considerando que as opções fornecidas não eram suficientes caso existisse um elevado número de problemas.

Em relação à junção de novos conjuntos de heurísticas a preferência foi claramente para o botão no menu do lado esquerdo. As justificações foram de que, se a opção estivesse inserida na “Combo Box”, seria ocupado muito espaço da interface e que, caso a solução fosse a última opção, se existirem muitos conjuntos de heurísticas em uso, tal não seria intuitivo.

Relativamente ao formato dos botões em geral as opiniões foram de que seria indiferente em termos de perceptibilidade, mas que esteticamente o formato de botão rectangular era mais agradável.

Quanto à opção de esconder a lista de heurísticas, todos os alunos consideraram que, apesar de ser um dos elementos mais importantes, deve existir uma opção para a sua ocultação para dar mais flexibilidade à aplicação. Quando questionados sobre a forma como tal poderia ser implementado, o consenso foi que um menu em formato “*Drop Down*” seria o mais adequado.

Finalmente, quanto à melhor forma de associar as violações de heurísticas à imagem, os alunos foram unânimes que a implementação de uma funcionalidade de “*Drag and Drop*” seria não só intuitiva, mas flexível.

Concluindo, embora o número de utilizadores tenha sido reduzido, o balanço final desta avaliação com base no protótipo de baixa fidelidade foi muito positivo tendo existido não só uma excelente reacção ao conceito, mas muito *feedback* para ser utilizado na implementação, nomeadamente no espaço alocado à imagem e nas opções com ela relacionadas.



## 4 – Implementação, Avaliação e Resultados

### 4.1 – Linguagem utilizada e considerações sobre a implementação

Durante a dissertação foram feitas várias opções sobre ferramentas a utilizar para o desenvolvimento da aplicação, desde *software* de prototipagem até à linguagem de programação utilizada. Nesta secção discutem-se as ferramentas e linguagem de programação utilizadas pretendendo dar uma noção do porquê das opções que foram tomadas.

#### 4.1.1 – Balsamiq Mockups 3

Bastante cedo no processo de desenvolvimento existiu uma necessidade de testar as funcionalidades a implementar numa interface de forma a poderem ser validadas. A interface tinha também de ser passiva de modificação rapidamente o que tornava inviável uma interface realizada em código. Ao mesmo tempo, pretendia-se testar a interactividade entre as várias interfaces pois foi um dos principais pontos estabelecidos aquando do *design* da aplicação. O Balsamiq Mockups 3 dava resposta a todas estas necessidades pelo que foi a ferramenta escolhida para a prototipagem de baixa fidelidade realizada ao longo da dissertação.

Balsamiq Mockups é uma ferramenta de *design* de *mockups* para interfaces de utilizador ou websites. Esta permite ao *designer* construir as suas interfaces através de *widgets* já existentes, utilizando métodos de *drag and drop* num editor específico.

O Balsamiq Mockups foi primariamente utilizado para a construção de um protótipo que foi depois avaliado com recurso a utilizadores. A sua fácil utilização e o facto de os *mockups* criados serem interactivos, permitindo facilmente ir de uma interface para a outra e assim testar todo o fluxo da aplicação, foram de extrema importância daí se ter optado por esta ferramenta.

### 4.1.2 – Linguagem C#

Uma das decisões mais importantes que é tomada durante o desenvolvimento de uma aplicação é a linguagem em que é desenvolvida. Conforme já foi demonstrado anteriormente, existiram várias versões conceptuais a pensar tanto em desenvolvimento web como em *desktop*. Era assim necessária uma linguagem com alguma flexibilidade e a capacidade de desenvolver para desktop e facilmente transpor esse design para web. A linguagem C# tornou-se assim uma escolha relativamente óbvia, reforçada ainda pela familiaridade já existente com a mesma.

A linguagem C# é uma linguagem de programação multiparadigma, possuindo paradigmas de programação imperativa, funcional e orientada a objectos. Foi desenvolvida pela Microsoft, incluindo influências de outras linguagens como Pascal e principal Java. É uma das linguagens que pode suportar a *framework* .NET.

Uma das principais características de C# é assim a capacidade de transpor de forma relativamente fácil o código desenvolvido para a *framework* .NET o que permite a criação de uma versão web de forma relativamente simples. Esta foi uma das principais razões pelas quais foi escolhida, mas também o facto de funcionar com Microsoft Visual Studio que facilita o processo de programação de interfaces de utilizador.

### 4.1.3 – Microsoft Visual Studio 2017

Após se ter optado pela linguagem C# a decisão do ambiente de desenvolvimento foi relativamente simples tendo-se optado pelo Microsoft Visual Studio.

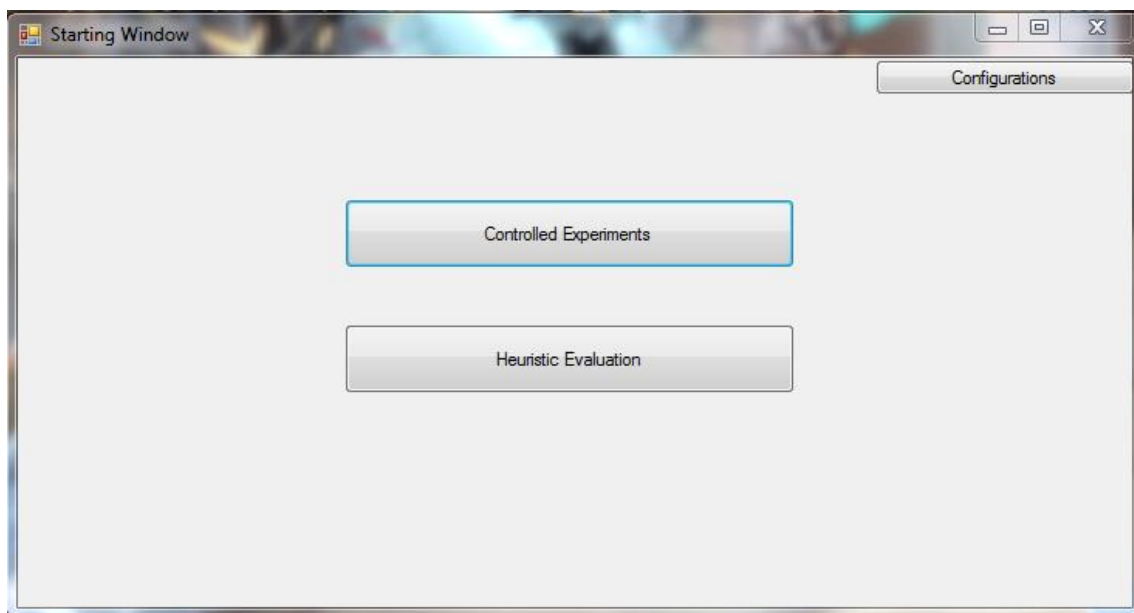
O Microsoft Visual Studio é um ambiente de desenvolvimento integrado criado pela Microsoft para desenvolvimento de software especialmente dedicado à *framework*.NET. É também o ambiente de desenvolvimento predilecto para a utilização da linguagem C#.

Para além de dar resposta às necessidades do projecto em termos de requisitos das linguagens utilizadas, o Microsoft Visual Studio possui também um editor de criação de interfaces com recurso a *drag and drop* que facilita bastante o processo de criação da interface utilizada na aplicação. Um dos principais problemas do Microsoft Visual Studio era a sua performance, sendo lento e com alguns *bugs*, mas a versão 2017 é uma melhoria significativa em relação às versões anteriores tendo uma performance bastante superior às mesmas.

## 4.2 – Implementação

Nesta secção irão ser mostradas as interfaces finais da aplicação, que poderão ser directamente comparadas aos protótipos desenvolvidos aquando do teste de baixa fidelidade. Serão também descritas as várias funcionalidades em cada interface e dado contexto sobre o porquê da sua implementação.

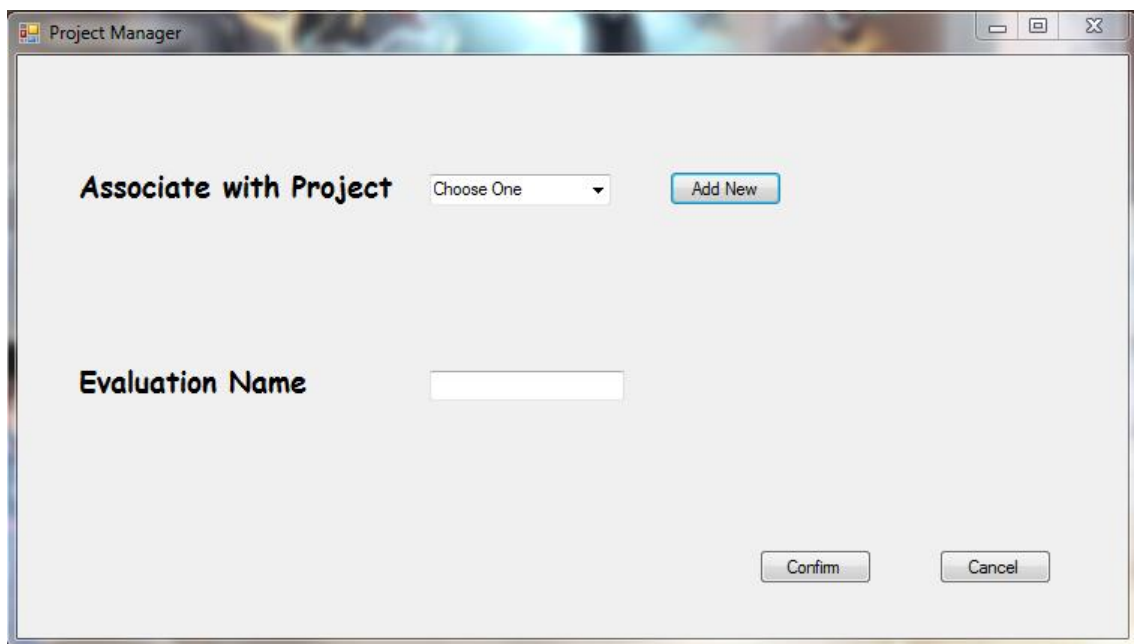
Na figura 15 é possível visualizar o primeiro ecrã da aplicação e, tal como no protótipo criado, existe uma separação dos dois métodos de avaliação suportados, sendo pedido ao utilizador que seleccione qual pretende aceder.



**Figura 15 - Ecrã inicial para escolha de uma das modalidades de avaliação previstas: Experiências Controladas ou Avaliação Heurística**

Na versão actual da aplicação apenas está disponível a avaliação heurística. O menu de configurações também não está implementado pois apesar de ter sido considerado necessário para a aplicação não é algo prioritário e não são ainda claras quais as configurações mais importantes e que devem ser implementadas.

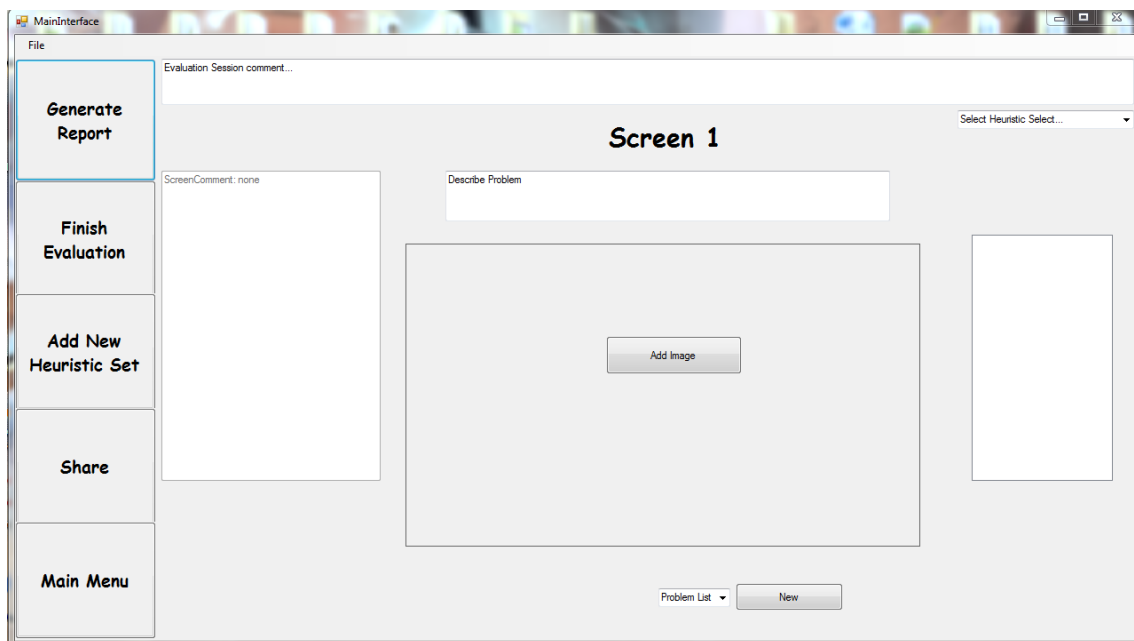
Após ser seleccionado o tipo de avaliação irá ser apresentado o “Project Menu “ (Figura 16).



**Figura 16-** Ecrã do protótipo do gestor de projectos do utilizador

Neste menu o utilizador pode associar a avaliação que irá criar com um projecto já existente ou criar um novo projecto. O objectivo do sistema de projectos é, como foi mencionado na fase de prototipagem, permitir que o utilizador tenha um sistema de organização próprio, onde os projectos funcionam de forma semelhante a directórios e onde as avaliações são os ficheiros. Também as funcionalidades associadas a este menu não foram implementadas devido a limitações de tempo e não tendo sido testada ainda a viabilidade ou utilidade do sistema.

Na figura 17 pode ser observada a área de trabalho principal da aplicação antes de qualquer imagem ou heurísticas associadas.



**Figura 17 - Menu Principal na sua forma mais básica sem heurísticas ou imagem**

Esta é constituída por uma área central englobando uma área para a imagem, número do *Screen*, descrição do mesmo bem como um “log” de problemas de usabilidade. No extremo direito da interface está a componente heurísticas que será discutida mais à frente, enquanto no extremo esquerdo estão funcionalidades que não estão relacionadas com a manipulação da imagem central. Existem também botões contendo operações relativas aos problemas heurísticos identificados, mas que apenas serão visíveis quando um problema esteja seleccionado de modo a facilitar a interacção do utilizador com a aplicação. Um exemplo da aplicação contendo já heurísticas, imagem e todos os menus visíveis pode ser observado na figura 18.

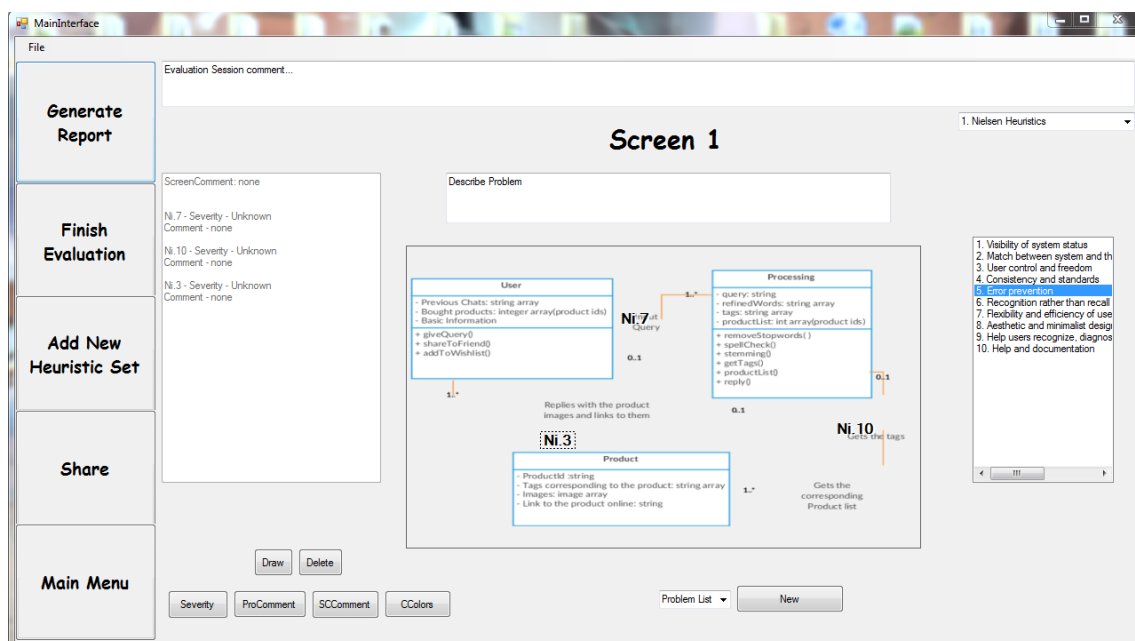


Figura 18 – Menu Principal após adicionar heurísticas e imagem

Logo, aqui é possível observar o claro foco em tentar que a área de trabalho tenha todas as funcionalidades separadas de forma lógica de modo a evitar a necessidade de muitas interfaces. Esta área irá ser explicada em maior detalhe ao longo das próximas páginas e descrito o fluxo de trabalho e funcionalidades disponíveis.

Para além das funcionalidades relacionadas com a escolha de heurísticas, a interface inclui ainda o acesso a outras opções, nomeadamente:

- *Generate Report* – Permite a criação de uma pasta contendo todas as imagens da avaliação bem como um ficheiro com o log dos vários problemas. Isto permite exportar a informação do programa para ser utilizada na criação de um relatório.

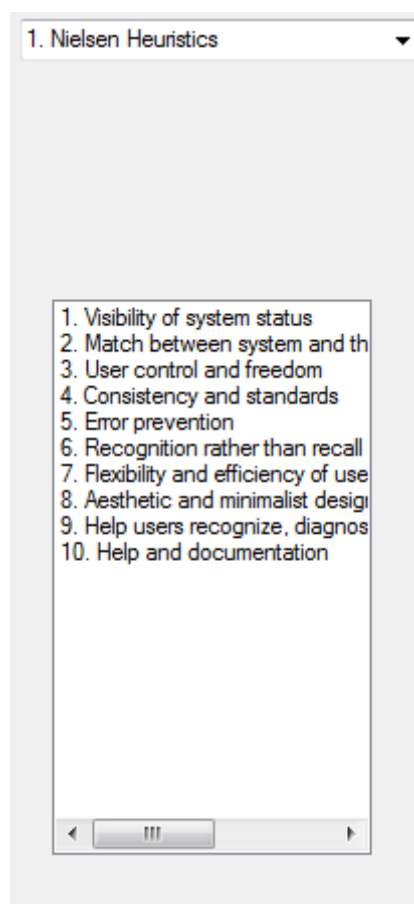
- *Finish Evaluation* – Esta opção serve para indicar que a avaliação em que se está actualmente a trabalhar está finalizada. Neste momento é ainda incerto se a aplicação deve ter um comportamento específico aquando da finalização da avaliação pelo que esta opção poderá ser removida.

- *Add New Heuristic Set* – Uma das opções mais importantes e parte do fluxo principal da aplicação. Esta opção permite adicionar conjuntos de heurísticas à aplicação que poderão posteriormente ser utilizadas para assinalar potenciais problemas na imagem escolhida. Existe uma grande flexibilidade nos conjuntos que podem ser adicionados, sendo possível adicionar conjuntos personalizados através de ficheiros XML desde que seja respeitado o formato suportado.

- *Share* – Funcionalidade que permite a partilha da avaliação actual. Esta foi uma das funcionalidades que não foi desenvolvida neste primeiro protótipo, mas permitirá ao utilizador fazer a partilha da avaliação actual, por exemplo como ilustração de uma avaliação heurística ou uma avaliação. De momento não se encontra implementada, existindo ainda dúvidas sobre a forma como essa partilha deve ocorrer, mas o que foi discutido está presente na forma de um *template*.

- *Main Menu* – Leva o utilizador para o primeiro menu da aplicação permitindo uma troca fácil entre avaliação heurística e experiência controlada.

Conforme pode ser observado na figura 19, após adicionar as heurísticas ao programa é possível seleccionar, de uma lista com os diferentes grupos de heurística carregados, o que se pretende, ficando as heurísticas visíveis e prontas a ser utilizadas, tendo o menu o aspecto mostrado na figura.



**Figura 19** - Menus relacionados com a escolha do grupo de heurísticas, no topo, e a lista das heurísticas que o compõem. Neste caso, em particular, foram seleccionadas as heurísticas de Nielsen

A partir deste momento é possível fazer “*drag and drop*” das heurísticas para a imagem. A notação criada utiliza os dois primeiros caracteres do nome do conjunto de heurísticas e o número da heurística, pelo que se fosse assinalado um problema de usabilidade que viola a terceira heurística de Nielsen a notação seria “Ni.3” (Figura 18). Para auxiliar os utilizadores na interpretação das heurísticas existe a opção de colocar o rato por cima das mesmas, o que irá mostrar uma “*tooltip*” contendo a descrição da heurística seleccionada. Estas descrições devem ser definidas no ficheiro XML que é utilizado para carregar heurísticas no programa.

Também na figura 18 é possível observar o *log* de problemas. Sempre que é assinalado um problema de usabilidade através de “*drag and drop*” é criada uma entrada no *log* de problemas de modo a dar ao utilizador um historial dos problemas encontrados. Note-se que não é possível editar este *log* de problemas.

É, no entanto, possível modificá-lo, atribuindo novos valores de severidade e comentários. Estas funcionalidades encontram-se disponíveis nos botões da parte inferior esquerda da interface (Figura 20) e apenas ficam disponíveis aquando da selecção de um problema de usabilidade. As mesmas funcionalidades estão também acessíveis através de um menu de contexto que pode ser acedido ao seleccionar o botão direito num problema de usabilidade.

Esta secção foi criada para trabalhar especificamente com os problemas de usabilidade, bastando para isso seleccioná-los e em seguida seleccionar a opção pretendida. É possível atribuir-lhes uma severidade, mudar a cor para melhor visibilidade em imagens diferentes ou fazer um comentário associado a esse mesmo problema (*ProComment*).

Existe também uma ferramenta básica de *draw* para permitir a criação de *highlights* e um botão de “*SCComment*” para atribuir um comentário ao *log* de problemas.

Existe ainda uma opção para fazer *delete* que foi adicionado devido a *feedback* por parte dos utilizadores, mas esta opção pode também ser feita através do menu de contexto ou através do clique do botão “*Delete*” do teclado.

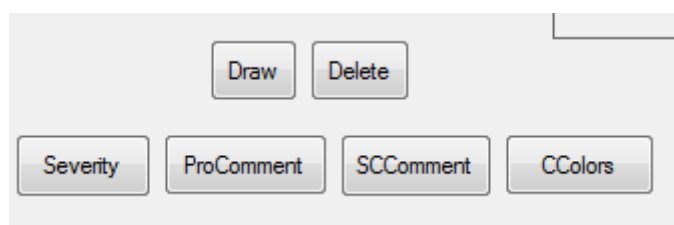


Figura 20 – Conjunto de opções disponíveis para um problema assinalado



Finalmente, existem opções de navegação e adição de outros problemas detectados pelo avaliador recorrendo à utilização de botões “*Next*” e “*Previous*”. Para casos extremos foi também criada uma “*Drop Down Box*” que permite acesso rápido ao problema desejado, sem necessidade de múltiplos cliques de “*Next*” ou “*Previous*”.

## 4.3 – Avaliação da aplicação

Após a construção de um protótipo já bastante completo existe a necessidade de testar as várias funcionalidades e a sua recepção por parte dos utilizadores, nomeadamente por um dos públicos-alvo que estavam contemplados: os estudantes.

Para realizar esta avaliação foram reunidos dez alunos de uma turma de IHC (Interação Humano Computador), do sexo masculino e feminino e idades compreendidas entre os 19 e os 22 anos para testarem a aplicação numa avaliação heurística formal.

Nesta avaliação foram criadas tarefas com o intuito de receber *feedback* com vista à melhoria da aplicação, bem como testar como os utilizadores reagiam a determinadas escolhas com as funcionalidades. Após as tarefas estarem completadas, existia uma conversa informal com o utilizador para que pudesse exprimir a sua opinião sobre a aplicação bem como dificuldades. Finalmente, os alunos realizavam o questionário SUS (*System Usability Scale*) para se obter uma visão global da usabilidade da aplicação (Brooke, 1996).

A avaliação da aplicação consistiu num conjunto de dez tarefas que visavam testar diferentes aspectos da aplicação. Os alunos eram encaminhados para uma sala com o avaliador e realizavam as tarefas propostas, podendo desistir das mesmas se desejassem, sendo cronometrados e avaliados quanto à facilidade com que completavam as tarefas. Após as tarefas estarem completadas, existia uma conversa informal com o aluno para que pudesse exprimir a sua opinião sobre a aplicação bem como dificuldades.

A lista de tarefas realizada pelos alunos foi a seguinte:

- 1- Adicionar uma imagem à aplicação;
- 2- Adicionar e seleccionar o *set* de heurísticas de Nielsen;
- 3- Na imagem, assinalar 3 potenciais problemas de usabilidade;
- 4- Atribuir gravidade a um dos problemas de usabilidade;
- 5- Alterar a cor desse problema;
- 6- Atribuir comentário ao *log* de problemas e um comentário específico a um segundo problema de usabilidade;
- 7- Eliminar o terceiro problema de usabilidade;
- 8- Adicionar o *set* de heurísticas de Zuk e mencionar quantas heurísticas fazem parte deste *set*;

- 9- Criar um novo potencial problema;
- 10- Regressar ao problema anterior.

As primeiras tarefas (1-3) tinham como intenção testar o fluxo principal da aplicação e perceber o quão intuitivo era para os utilizadores, sendo este um dos pontos mais importantes ao nível do *design*. As tarefas seguintes (4-7) visavam testar as funcionalidades existentes na *toolbar* e que são referentes aos problemas de usabilidade criados na imagem. Finalmente as restantes tarefas (8-10) estavam relacionadas com a selecção de novas heurísticas e navegação na aplicação.

À medida que iam realizando as tarefas eram avaliados os seguintes parâmetros:

- 1- Número de cliques;
- 2- Completou a tarefa com sucesso;
- 3- Tempo que levou a completar (máximo 2 minutos);
- 4- Cometeu erros;
- 5- Sentiu-se perdido;
- 6- Solicitou ajuda;
- 7- Grau de dificuldade observada de 1 a 5 sendo 1 o mais difícil e 5 o mais fácil.

Estas medidas permitem ter uma ideia do quão difícil uma tarefa é de completar bem como de observar e se mesmo quando completada com sucesso essa tarefa gerou dúvidas nos utilizadores. Tudo isto permite uma boa recolha de *feedback* mostrando onde existem fluxos ou funcionalidades que podem ser melhorados.

Finalmente, o preenchimento do inquérito SUS (em Anexo), que avalia aspectos como a curva de aprendizagem, pertinência das funcionalidades ou a confiança do utilizador durante a utilização da aplicação, permitia perceber qual era a opinião dos alunos quanto à usabilidade, tendo ainda a vantagem de ser um questionário creditado com resultados e fiabilidade já comprovados. Ainda que exista uma versão do SUS em português, da qual já foi feita uma primeira validação (Martins et al., 2015), a utilização da escala em inglês não foi considerada um obstáculo para o público alvo.

## 4.4 – Discussão dos Resultados

Após realizada o teste com utilizadores foi necessário analisar os resultados obtidos. No geral foram bastante satisfatórios demonstrando que a decisão de centrar o desenvolvimento no utilizador compensou.

No fluxo principal do programa (tarefas 1-3) não existiram grandes dificuldades tendo todos os utilizadores completado as tarefas e na sua maioria completaram cada tarefa em menos de 15 segundos, tendo existido apenas alguma confusão inicial quando foi necessário adicionar problemas de usabilidade por *“drag and drop”*. Este é um resultado bastante positivo indicando que os principais componentes da aplicação são fáceis de aprender e utilizar.

Nas funcionalidades secundárias (tarefas 4-7) os resultados foram também bastante positivos com os utilizadores a utilizarem as funcionalidades secundárias com relativa facilidade. No entanto, apagar um problema de usabilidade revelou-se problemático para alguns alunos. A forma como essa remoção é feita é clicando na tecla *“delete”* após o problema estar seleccionado. Alguns alunos rapidamente chegaram à solução, mas a grande maioria procurou solução na interface em botões ou menus de contexto quando tal opção não existia. Os alunos indicaram que toda a aplicação era *“mouse driven”* e que após estarem habituados a operá-la com o rato, ao depararem-se com uma funcionalidade apenas em teclado ficaram confundidos. Esta crítica foi tida em conta e resultou num menu de contexto, algo que os utilizadores tentaram encontrar na aplicação anteriormente, para a remoção de problemas de usabilidade na versão final. Nas tarefas finais de navegação também os resultados foram excelentes, não tendo os alunos qualquer dificuldade em completá-las.

Também no teste de SUS os resultados foram bastante positivos. Após os cálculos, os valores variam entre os 70 e os 95 pontos o que é extremamente positivo considerando que uma aplicação ter um valor superior a 68 no SUS já é considerada uma boa aplicação do ponto de vista da usabilidade (Bangor et al., 2008; Brooke, 2013). A média das pontuações foi de 80.3 pontos e é de salientar também que a maioria das avaliações foi realizada por utilizadores sem qualquer contacto anterior com a aplicação, confirmando os resultados observados durante a realização das tarefas.

Os resultados finais são assim bastante animadores apontando para que as metas estabelecidas ao nível da intuitividade e usabilidade tenham sido atingidas. Há, no entanto, que dizer que o número de alunos que testaram foi relativamente pequeno pelo que será necessário um maior número de utilizadores para testar a aplicação de forma a confirmar definitivamente os resultados obtidos.



## 5 – Conclusões e Trabalho Futuro.

A avaliação em Visualização de Informação é muito importante para utilizadores e investigadores. No entanto, é muitas vezes ignorada, embora existam muitos métodos de avaliação que podem ser usados neste contexto, não foi possível encontrar ferramentas de avaliação específicas para responder às necessidades de quem trabalha nesta área.

Foi objectivo desta dissertação contribuir para mitigar este problema através da criação de uma aplicação desenvolvida especificamente de raiz para responder às necessidades de avaliação em Visualização de Informação utilizando métodos de avaliação em geral bastante usados na avaliação de usabilidade: avaliação heurística, questionários e experiências controladas, identificados num estudo inicial como sendo adequados ao fim em vista.

O desenvolvimento da aplicação envolveu pesquisa não só ao nível do mercado e das funcionalidades mas também sobre o público-alvo, tendo sido desde cedo adaptada a filosofia de desenvolvimento centrado no utilizador para um maior foco no utilizador, na usabilidade e na constante realização de testes ao longo das várias fases de desenvolvimento.

Foram identificadas lacunas existentes para a avaliação de sistemas interactivos, nomeadamente na aplicação de sistemas de avaliações heurísticas, tendo sido também identificados os métodos mais promissores e integráveis no ciclo de desenvolvimento e investigação, tendo o resultado final sido uma aplicação mais facilmente integrável no mesmo, resultado de um modelo conceptual devidamente testado tanto em protótipo de baixa fidelidade como em protótipo funcional, ambos com avaliação positiva por parte dos utilizadores.

Ainda assim, existe algum trabalho a realizar nomeadamente um leque de testes mais alargados e com maior número de utilizadores e, ainda que tenham sido incluídos contributos de elementos considerados representativos de investigadores, uma avaliação formal com este grupo de utilizadores seria um contributo importante para informar e validar o trabalho futuro. É necessária, também, a inclusão do suporte para experiências controladas, sendo necessário um processo semelhante ao que foi utilizado para a criação da componente de avaliação heurística, tendo sido um dos dois métodos de avaliação discutidos ao longo de toda a dissertação como passível de ser implementado durante os processos de desenvolvimento de *software*. Por último é necessário mencionar que o tipo de cenários equacionados justifica o desenvolvimento de uma aplicação *web* no futuro.



## Referências

- Adlin, T., & Pruitt, J. (2010). *The essential persona lifecycle: Your guide to building and using personas*. Morgan Kaufmann Publishers Inc.
- Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the system usability scale. *Intl. Journal of Human-Computer Interaction*, 24(6), 574-594. <http://dx.doi.org/10.1080/10447310802205776>
- Brooke, J. (1996). SUS: a "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & A. L. McClelland (eds.). *Usability Evaluation in Industry*. London: Taylor and Francis, 189–194
- Brooke, J. (2013). SUS: a retrospective. *Journal of usability studies*, 8 (2), 29-40.
- Cooper, A., Reimann, R., and Cronin, D. (2007). *About Face 3: The Essentials of Interaction Design*. John Wiley & Sons, Inc.
- Blandford, A., Cox, A., & Cairns, P. (2008). Controlled experiments. *Research Methods for Human-Computer Interaction*, 1–16.
- Carpendale, S. (2008). Evaluating Information Visualizations. *Information Visualization, Human-centered issues and perspectives*. Springer, 19–45. [https://doi.org/10.1007/978-3-540-70956-5\\_2](https://doi.org/10.1007/978-3-540-70956-5_2)
- Dix, A., Finlay, J., Abowd, G. D., & Beale, R. (2004). *Human-Computer Interaction*, 3<sup>rd</sup> ed, Prentice Hall.
- Fonseca, J., Campos, P., Gonçalves, D. , (2012). *Introdução ao Design de Interfaces*. FCA.
- Forsell, C., & Johanson, J. (2010). An heuristic set for evaluation in information visualization. In *Proceedings of the International Conference on Advanced Visual Interfaces AVI2010*, 199–206. <https://doi.org/10.1145/1842993.1843029>
- Freitas, C., Luzzardi, P. R. G., Cava, R. A., Winckler, M. A. A., Pimenta, M. S., & Nedel, L. P. (2002). Evaluating Usability of Information Visualization Techniques. In *Proc Symposium on Human Factors in Computer Systems, IHC 2002*.

- Freitas, C., Pimenta, M. S., & Scapin, D. (2009). User-Centered Evaluation of Information Visualization Techniques : Issues and Perspectives. *Anais do Colóquio em Informática: Brasil / INRIA. Cooperações, Avanços e Desafios*, 2603–2606. [http://doi.org/10.1007/978-1-4614-7485-2\\_12](http://doi.org/10.1007/978-1-4614-7485-2_12)
- Gómez, R. Y., Caballero, D. C., & Sevillano, J. (2014). Heuristic Evaluation on Mobile Interfaces: A New Checklist, *The Scientific World Journal*, Article ID 434326, 19 pages. <http://dx.doi.org/10.1155/2014/434326>
- Hermawati, S., Lawson, G. (2016). Establishing usability heuristics for heuristics evaluation in a specific domain: Is there a consensus? *Applied Ergonomy*, 56, 34-51. <http://doi.org/10.1016/j.apergo.2015.11.016>
- Khowaja, K., & Salim, S. S. (2015). Heuristics to Evaluate Interactive Systems for Children with Autism Spectrum Disorder (ASD), *PLOS One*, 1–27. <http://doi.org/10.1371/journal.pone.0132187>.
- Martins, A. I., Rosa, A. F., Queirós, A., Silva, A., & Rocha, N. P. (2015). European Portuguese validation of the System Usability Scale (SUS). *Procedia Computer Science*, 67, 293-300.
- Mayhew, D. J. (1999). *The Usability Engineering Lifecycle: A Practioner's Handbook for User Interface Desig*. Morgan Kaufmann.
- Miaskiewicz, T., Kenneth, A. K. (2011). Personas and user-centered design: How can personas benefit product design processes? *Design Studies*, 32(5), 417-430. <https://doi.org/10.1016/j.destud.2011.03.003>
- Nielsen, J. (1993). *Usability Engineering*, Morgan Kaufmann.
- Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* 152–158.
- Plaisant, C. (2004). The Challenge of Information Visualization Evaluation. *Proceedings of the working conference on Advanced visual interfaces AVI 2004*, 109-116. <https://doi.org/10.1145/989863.989880>
- Rudd, J., Stern, K., & Isensee, S. (1996). Low vs. high-fidelity prototyping debate. *Interactions*, 3(1), 76-85. <https://doi.org/10.1145/223500.223514>
- Roth, S. F., Lucas, P., Senn, J. A., Gomberg, C. C., Burks, M. B., Stroffolino, P. J., ... Dunmire, C. (1996). Visage: A User Interface Environment for Exploring Information. *Proceedings IEEE Symposium on Information Visualization '96*, 3–12. <https://doi.org/10.1109/INFVIS.1996.559210>



- Santos, B.S., Ferreira, B.Q.,Dias, P. (2016). Using Heuristic Evaluation to Foster Visualization Analysis and Design Skills. *IEEE Computer Graphics and Applications*, 36 (1), 86 – 90. <https://doi.org/10.1109/MCG.2016.7>
- So, C., & Joo, J. (2017). Does a Persona Improve Creativity ? Does a Persona Improve Creativity ? *The Design Journal*, 20(4), 459–475. <http://doi.org/10.1080/14606925.2017.1319672>
- Zuk, T., Schlesier, L., Neumann, P., Hancock, M. S., & Carpendale, S. (2006). Heuristics for Information Visualization Evaluation. In *First Workshop on Beyond Time and Errors Novel Evaluation Methods for Visualization BELIV'06*, 1–6. <http://doi.org/10.1145/1168149.1168162>

## **Anexos**

Ao longo do processo de desenvolvimento da dissertação foi realizada uma avaliação da aplicação recorrendo a um conjunto de participantes. Essa avaliação incluía um conjunto de tarefas, guias de orientação específicos e culminou com a aplicação de um questionário SUS (*System Usability Scale*). Essa documentação encontra-se presente nesta secção

# Guia do Observador:

NI de utilizador: \_\_\_\_\_ Guião **do Observador**

Tarefa	N° cliques	Completo <u>u</u> a Tarefa?	Tempo Máximo Tempo observado (mm:ss)	Cometeu erros?	Sentiu-se perdido?	Solicitou ajuda	Grau de facilidade observada 1 – Nada Fácil 5 – Muito Fácil					
1		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
2		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
3		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
4		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
5		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
6		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
7		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
8		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
9		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								
10		não <input type="checkbox"/> sim <input type="checkbox"/>	2m :	não <input type="checkbox"/> poucos <input type="checkbox"/> muitos <input type="checkbox"/>	não <input type="checkbox"/> pouco <input type="checkbox"/> muito <input type="checkbox"/>	não <input type="checkbox"/> sim <input type="checkbox"/> qual?	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5								

Observações

Figura 1 – Guia do Observador

## Guia de Tarefas do Utilizadores:

### *Teste de usabilidade*

#### *sobre aplicação para apoio à realização de avaliação heurística*

Agradecemos a sua colaboração na realização deste estudo, que tem por objectivo avaliar a aplicação. A sua colaboração constitui um factor importante para o êxito desta avaliação, por isso solicitamos-lhe a realização destas tarefas. A participação neste teste de usabilidade é voluntária e pode cessá-la a qualquer momento. Toda a Informação recolhida é estritamente confidencial, no entanto os resultados poderão ser usados para melhorar o sistema. O seu nome ou qualquer outro aspecto da sua identificação não serão usados em caso algum.

**Note que não está a ser avaliado! A aplicação é que está a ser avaliada.**

### **Teste 1:**

<b>Tarefa 1</b>	Adicione uma imagem. -----
-----------------	-------------------------------

<b>Tarefa 2</b>	Adicione e seleccione o set de heurísticas de Nielsen -----
-----------------	--

<b>Tarefa 3</b>	Na imagem, assinale 3 potenciais problemas de usabilidade -----
-----------------	--

<b>Tarefa 4</b>	Atribua uma gravidade de um dos problemas de usabilidade -----
-----------------	---

<b>Tarefa 5</b>	Mude a cor a esse problema -----
-----------------	-------------------------------------

<b>Tarefa 6</b>	Atribua um comentário livre ao log de problemas. De seguida atribua um comentário específico a um segundo problema de usabilidade -----
-----------------	--

<b>Tarefa 7</b>	Apague o terceiro problema -----
-----------------	-------------------------------------

<b>Tarefa 8</b>	Adicione e adicione o set de heurísticas de Zuk. Quantas heurísticas existem? -----
-----------------	--

<b>Tarefa 9</b>	Crie um novo "problema" -----
-----------------	----------------------------------

<b>Tarefa 10</b>	Volte ao problema anterior -----
------------------	-------------------------------------

## Questionário SUS:

### *System Usability Scale*

© Digital Equipment Corporation, 1986.

	Strongly disagree							Strongly agree
1. I think that I would like to use this system frequently	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
2. I found the system unnecessarily complex	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
3. I thought the system was easy to use	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
4. I think that I would need the support of a technical person to be able to use this system	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
5. I found the various functions in this system were well integrated	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
6. I thought there was too much inconsistency in this system	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
7. I would imagine that most people would learn to use this system very quickly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
8. I found the system very cumbersome to use	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
9. I felt very confident using the system	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			
10. I needed to learn a lot of things before I could get going with this system	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	1	2	3	4	5			

Figura 2 – Questionário SUS (System Usability Scale)